

DEVELOPMENT OF A FINITE ELEMENT PROGRAM
INCORPORATING ADVANCED ELEMENT TYPES

By

RAMEEZ IQBAL

Bachelor of Engineering

Osmania University

Hyderabad, India

2007

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
July, 2009

DEVELOPMENT OF A FINITE ELEMENT PROGRAM
INCORPORATING ADVANCED ELEMENT TYPES

Thesis Approved:

Dr. Jonathan S. Goode

Thesis Adviser

Dr. Robert N. Emerson

Dr. Bruce Russell

Dr. A. Gordon Emslie

Dean of the Graduate College

ACKNOWLEDGMENTS

I thank God for guiding me to the right path.

I express my deep gratitude and appreciation to Dr. Jonathan S. Goode for his confidence in me towards this project. His patience, guidance and constant encouragement helped me immensely to complete my research project. I am indebted for all the time he has provided for my research project.

I thank Dr. Robert Emerson for his consistent support and assistance concerning various aspects of my study. His invaluable experience helped immensely in the development of this project.

I also thank Dr. Bruce Russell for his encouragement and guidance rendered to me throughout my research.

I am deeply grateful to my friend and colleague Muhammet A. Saglar for his constant support during the successful completion of this thesis. I am really grateful to him for accommodating the extra workload.

I thank my family and friends for the enormous support they have shown me.

And finally, I thank my parents for their unconditional support and encouragement I received during the past two years. I thank them from the bottom of my heart the values they instilled in me for hard work, honesty and dedication.

TABLE OF CONTENTS

Chapter	Page
I. INTRODUCTION	1
1.1 Background and Problem Definition	1
1.2 Objectives	3
1.3 Overview	4
II. LITERATURE REVIEW	6
2.1 Finite Element Method	6
2.2 Advanced Elements	7
2.3 Software Advantages	10
III. FINITE ELEMENT THEORETICAL DERIVATIONS	12
3.1 Introduction	12
3.2 Interpolation and Shape Functions	13
3.3 Formulation of Element Stiffness Matrices	14
3.4 Element Boundary and Loading Conditions	16
3.5 Solution Techniques	17
3.6 Advanced Finite Element Formulations	18
3.6.1 Isoparametric Elements	18
3.6.2 Constant Strain Triangular Elements (CST)	21
3.6.3 Linear Strain Triangular Elements (LST)	24
3.6.4 Bilinear Rectangular Elements (Q4)	27
3.6.5 Quadratic Rectangular Elements (Q8)	29
IV. CASE STUDIES	33
4.1 Introduction	33
4.2 CST Elements	35
4.3 LST Elements	38
4.4 Q4 Elements	42
4.5 Q8 Elements	45
4.6 Comparison	49

Chapter	Page
V. CONCLUSION	51
5.1 Summary and Conclusions	51
5.2 Recommendations	52
REFERENCES	54
APPENDIX A – MATLAB FEA PROGRAM.....	56
APPENDIX B – CONFIRMATION CALCULATIONS OF ELEMENT FORMULATIONS	65

LIST OF TABLES

Table	Page
4-1 Input Values for Various Elements.....	35
4-2 Maximum Tip Deflection for CST Elements.....	38
4-3 Maximum Tip Deflection for LST Elements.....	41
4-4 Maximum Tip Deflection for Q4 Elements.....	45
4-5 Maximum Tip Deflection for Q8 Elements.....	49
4-6 Maximum Tip Deflection Comparison.....	50

LIST OF FIGURES

Figure	Page
3-1 Four Node Isoparametric Element	20
3-2 Three Node CST Element	23
3-3 Six Node LST Element	26
3-4 Four Node Q4 Element	28
3-5 Eight Node Q8 Element	31
4-1 Fixed Beam Loaded at the Free End	34
4-2 CST Element Mesh	36
4-3 CST Beam Deflection Profile	36
4-4 Contour Plot for CST Element Displacements in the x -Direction	37
4-5 Contour Plot for CST Element Displacements in the y -Direction	37
4-6 LST Element Mesh	39
4-7 LST Beam Deflection Profile	40
4-8 Contour Plot for LST Element Displacements in the x -Direction	40
4-9 Contour Plot for LST Element Displacements in the y -Direction	41
4-10 Q4 Element Mesh	43
4-11 Q4 Beam Deflection Profile	43
4-12 Contour Plot for Q4 Element Displacements in the x -Direction	44
4-13 Contour Plot for Q4 Element Displacements in the y -Direction	44
4-14 Q8 Element Mesh	46
4-15 Q8 Beam Deflection Profile	47
4-16 Contour Plot for Q8 Element Displacements in the x -Direction	47
4-17 Contour Plot for Q8 Element Displacements in the y -Direction	48

CHAPTER I

INTRODUCTION

1.1 Background and Problem Definition

Finite element analysis (FEA) is one of the most common methods used for numerical solution of field problems. The field problem is often described as a differential equation or an integral expression. The finite element solution helps in solving these complex field problems into an easier ordinary differential equation that can then be numerically solved.

Finite element analysis can be used in any field problem such as heat transfer, stress/strain analysis of structural or mechanical systems, fluid flow, and magnetic fields. It can use any type of geometrical shape avoiding any restrictions in geometry. Boundary conditions, loading and material properties can vary from element to element. It can incorporate more than one type of element which closely resembles the actual body or the region which is to be analyzed.

At present many commercial types of software have been developed to analyze structures using finite elements. Software titles such as Staad.Pro, Ansys and Abaqus are some of

the commercial programs available for research and design purposes. These programs are usually advantageous in checking the element shapes, element meshes and connections. In the present study, however, Matlab is used in the development of a software program for finite element analysis.

Each element is connected at points known as nodes. In a two-dimensional analysis of displacements, each node has two degrees of freedom. The particular arrangement of the nodes is known as the finite element mesh. A finite element mesh is represented numerically by a system of algebraic equations to be solved for the unknown field quantities at the nodes. These unknowns are the values of the field quantity and depend on the element type. The field quantity over the entire structure is approximated element by element.

Different types of elements are used in this present study to analyze the beam for deflections. Subsequently, stresses and strains may also be determined. The stresses and strains, however, are not the focus of this study. As such, only deflections will be of interest herein. The different types of elements considered in this study are:

1. Constant Strain Triangular elements (CST),
2. Linear Strain Triangular elements (LST),
3. Four-Node Rectangular elements (Q4), and
4. Eight-Node Rectangular elements (Q8).

CST and LST elements are advantageous in determining the heat conduction and modeling pure bending of beams. Whereas, the Q8 elements help in eliminating shear locking in beam analysis that is experienced when using Q4 elements. Each of these elements can also be further used in formulating frames and trusses. Due to their flexibility, they can also be used to analyze structural dynamics due to earthquake ground motion as well as heat transfer and temperature degradation due to structural fire conditions.

In the present study, however, only two-dimensional (2-D) structural analysis of simple beams was considered. The finite element analysis program then consisted of 2-D elements to derive the structural equations. With known boundary conditions (or supports) and known loading conditions, the unknown field quantity (deflections) were then solved. The accuracy of the finite element solution depends on many factors. However, the primary focus of this study is not concerned with the individual accuracy of any single problem.

1.2 Objectives

This study incorporates the use of basic and advanced finite elements as a tool to solve complex engineering problems. The use of commercial software is also an important aspect in the present day research scenario to analyze the problem with limited difficulty. The primary objective of this study is the development of a finite element analysis (FEA) program, incorporating each of the four elements as previously defined, written in the

Matlab programming language. This program is capable of formulating the element stiffness matrix for all four types of elements. This program uses static and dynamic loading to solve for deflections, stress and strains. This program is also capable of generating graphical outputs for the results.

This study also helps in getting a better understanding of the various computational software being used for research and commercial purposes. In the field of Civil Engineering, finite element analysis has played an enormous role in solving complex analytical problems. It has drastically improved the economical part of this field by reducing the cost of laboratory tests for practical field problems.

1.3 Overview

The next several chapters present the methodology behind this study. A summary of each chapter is provided to give a brief overview of the remaining sections of this study:

- **Chapter II – Literature Review:**

Studies conducted on various disciplines using finite element analysis are presented and a brief overview of these studies is provided. This chapter also discusses the usefulness of software for research purposes and its ease in determining or extracting specific and relevant data to the problem being considered.

- **Chapter III – Finite Element Theoretical Derivations:**

Detailed theoretical development of the finite element analysis procedure is presented and discussed. The development of the structural equations for a structural system using finite elements is first presented. Application of boundary and loading conditions as well as solution techniques to determine the unknown deflections are presented. Finally, advanced finite element formulations considering the four element types considered in this study are presented. Each of these formulations are used in determining the structural equations for the system.

- **Chapter IV – Case Studies:**

The capabilities of the Matlab FEA program developed as a consequence of this study are demonstrated. Each of the four element types considered in this study are utilized to demonstrate the difference in accuracy between the elements. All the results that are needed for this study are provided in this chapter. This chapter also compares the results that are found from hand calculations done using the procedure of FE.

- **Chapter V – Conclusion:**

A summary of the results and conclusions of the study are provided. Recommendations are provided for potential future research with respect to the development of the FEA program as developed in this study.

CHAPTER II

LITERATURE REVIEW

2.1 Finite Element Method

The finite element method can be used to solve complex linear and non-linear problems. There are thousands of studies that have been conducted in the past ten years that could be considered. All of these studies together, demonstrate the flexibility for which the finite element method can be used. Several specific examples are provided herein that demonstrate this flexibility as it pertains to structural analysis as considered in this study.

A finite element code was developed for the curing simulation of thick composite structures by Park et al. (2000). This model was capable of modeling composite structures with arbitrary shapes. The proposed degeneration method to build the thermal conductivity matrix for two-dimensional cure simulation yielded satisfactory results. It also provided accurate temperature distributions during the curing of thick composite structures.

A finite element analysis was conducted on steel fiber-reinforced concrete beams by D. Mehmet et al. (2008). This beam was modeled by using nonlinear material properties

adopted from experimental study. Eight-node solid brick elements were used to model the concrete. The deflection and stress data were compared to the experimental study at failure and yielded the desired results. They found the initial and progressive cracking due to deflections and stress obtained from the finite element model compared well to the experimental data. They also found that the failure mechanism of the beam is modeled quite well using the finite element method and the failure load predicted was very close to the failure load measured during experimental testing.

Finite element method can also be used in frame structures. M. Barbato et al. (2006) analyzed the response sensitivity analysis using three-field mixed formulations for frame structures. The methodology was based on direct differentiation methods and produced the response sensitivities consistent with the numerical finite element response. This methodology is also related to the developed steel-concrete composite frame element. Particular attention was given to steel-concrete composite frame finite elements for which the three-field mixed formulation was found to be beneficial in terms of numerical accuracy. Furthermore, finite element response sensitivity analysis was found to give an adequate insight into the structural response behavior and its sensitivity to modeling parameters.

2.2 Advanced Elements

Advanced finite elements have also been used in thousands of studies in the past ten years. These elements go beyond the basic axial/bar and beam elements that are

commonly used in structural analysis (e.g., direct stiffness method). Incorporation of advanced elements allows the user to consider localized changes in geometry or material that may change the characteristics of the structural response. As such, the use of advanced elements provides superior flexibility with respect to a detailed analysis of the performance of the structural system. As before, a few select studies are discussed herein that pertain to the use of advanced elements in a finite element analysis.

Salem et al. (2001) developed a robust element distortion metric based on the concept of mid-node admissible spaces for two-dimensional (2-D) quadratic triangular finite elements with one edge curved. The concept implies that for any mid-node of a 2-D quadratic triangular element there is a specific area within which it can be placed such that element quality is maintained. This metric is based on the Jacobian matrix and requires computation of the linear element only. This process reduces the CPU time as calculated from the Jacobian values at gauss integration points.

Glaucio et al (2007) proposed a Q4/Q4M element having design variable nodes and displacement nodes which are not coincidental. They implemented this element using different approaches, including continuous approximation of material distribution and nodal approaches. In both the cases, it was found that the method used was more advantageous in generating higher resolution for topological results. They also applied an integral averaging technique to suppress the instabilities caused due to layering and islanding.

Li et al. (2001) presented a quadratic finite element and quadratic finite strip with generalized degrees of freedom based on the fact that the local displacement fields of the elements should be compatible with the global displacement field for the corresponding system. Though quadratic elements and strips were used, they found results with good accuracy and desirable convergence. Compared to the traditional finite elements and strips this method yielded similar results with less degrees of freedom. They also found that, when compared to the linear element, this method can yield results with better accuracy.

Kikuchi et al. (1999) presented a modification of an 8-node quadrilateral element which is widely used in finite element analysis. They proposed this element which can represent any Cartesian and isoparametric quadratic polynomials when it is of bilinear isotropic shape. They found that the results were in good agreement with the basic formulation of an 8-node element. Moreover, this element gave good results for higher order elements and for three-dimensional (3-D) elements.

Finally, Long et al. (2004) investigated the effect of modified reduced quadrature rules on the presence of spurious modes in the stiffness matrices of the Q8 serendipity and Q9 Lagrange membrane finite element. The alternative five- and eight-point schemes were proposed for Q8 and Q9 elements, respectively, that allowed for the elimination of spurious modes while element accuracy was maintained. They found that the Q8 element yielded more economical results using the five-point rule when compared to the eight-point rule. The Q9 element, however, produced inadequate results using the five-point

rule whereas the eight-point rule gave accurate results. Each of these results was compared to the fully integrated counterpart.

2.3 Software Advantages

The use of software to obtain finite element solutions has also been well documented through the literature. Commercial software packages are readily available that provide adequate flexibility for most research and practical purposes. However, the development of a FEA program through the use of computer programming provides superior flexibility with respect to user manipulation. User-written FEA programs have always been at the heart of research and have been utilized for many years at academic institutions. The programming language Matlab has become increasingly popular due to its capabilities with respect to matrix manipulation. A few examples are provided here that demonstrate the use of Matlab as the programming language for the development of a FEA program.

Alberty et al. (2002) presented a short Matlab implementation for P_1 and Q_1 finite elements for the numerical solution of 2-D and 3-D problems in linear elasticity with mixed boundary conditions. They provided a Matlab code incorporating the simple model examples as well as more complex problems. Element stiffness matrices were assembled by Dirichlet and Newmann methods for quadrilateral and tetrahedral elements. The authors provided the numerical examples with postprocessing and error estimation via an average stress field illustrating the new Matlab tool and its flexibility.

Barlets et al. (2004) implemented a short Matlab program to incorporate a flexible isoparametric finite element method. Two-dimensional domains with curved boundaries of elastic problems having quadratic order were considered. They incorporated triangular and quadrilateral elements equipped with varying quadrature rules which allowed for mesh refinement. They provided numerical examples for the Laplace equation with mixed boundary conditions to indicate the flexibility of the isoparametric finite elements.

CHAPTER III

FINITE ELEMENT THEORETICAL DERIVATIONS

3.1 Introduction

The primary objective of this study is to develop a finite element analysis program utilizing advanced finite elements. These different element types, however, influence the response of a beam in varying fashions. This effect depends on various factors such as the number of nodes per elements, the degrees of freedom associated with that element, the displacement field and the material properties. As such, one element is not always superior to another with respect to any given analysis. Often, it is the experience of the finite element analyst that determines the appropriate element to be used. The purpose of this study is not to determine the appropriate element to be used in an analysis. The purpose of this study is only to develop the capabilities for the analyst to choose the appropriate element.

To develop the finite element analysis program, an understanding of the theoretical development of finite elements is necessary. Finite elements are discrete pieces of the system that are interconnected at nodal points. In a structural sense, each element contributes to the stiffness of the system. The stiffness, in conjunction with the boundary

(or support) conditions as well as prescribed loadings, determines the deformations of the system. Numerical procedures are utilized to determine the stiffness for each element. Furthermore, algorithms for combining each element into an assembly of finite elements is needed to determine the structural stiffness. Finally, solution techniques are needed to solve the structural equations from a numerical perspective.

3.2 Interpolation and Shape Functions

Interpolation is to form a continuous function that satisfies prescribed conditions at a finite number of points. The points are the nodes of the elements and the prescribed conditions are the nodal values of the field quantity. The interpolating function is typically a polynomial that provides a single-valued and continuous field. Therefore, the generalized degree-of-freedom (DOF) a_i can be interpolated with dependent variable \emptyset and independent variable x that can be written in the form

$$\emptyset = \sum_{i=0}^n a_i x^i \quad \text{or} \quad \emptyset = [X]\{a\} \quad (3-1)$$

where

$$[X] = [1 \ x \ x^2 \ \dots \ x^n] \text{ and } \{a\} = [a_0 \ a_1 \ a_2 \ \dots \ a_n]^T \quad (3-2)$$

where $n = 1$ for linear interpolation, $n = 2$ for quadratic interpolation, and so on. The relationship between nodal value $\{\emptyset_e\}$ and a_i is given by

$$\{\emptyset_e\} = [A]\{a\} \quad (3-3)$$

where each row of $[A]$ is evaluated at the appropriate locations in $[X]$. Solving equations 3-1, 3-2 and 3-3 gives

$$\emptyset = [N]\{\emptyset_e\} \quad \text{where} \quad [N] = [X][A]^{-1} = [N_1 \ N_2 \ \dots] \quad (3-4)$$

where $[N]$ is a vector containing each shape function N_i . For each N_i , the shape function is unity at its own node and zeros at all other nodes. Here individual N_i in matrix $[N]$ is called the shape function.

Lagrange's interpolation formula provides a general form of shape functions which can be used in a variety of different types of elements. The formula is given as

$$N_k = \frac{(x_1 - x_2)(x_2 - x) \dots [x_k - x] \dots (x_n - x)}{(x_1 - x_k)(x_2 - x_k) \dots [x_k - x_k] \dots (x_n - x_k)} \quad (3-5)$$

For linear interpolation, shape functions and the coordinates do not have subscripts greater than 2, whereas for quadratic interpolation the subscripts are not greater than 3.

3.3 Formulation of Element Stiffness Matrices

Different mathematical procedures can be used to determine the element stiffness matrices. Virtual work, slope deflection, moment distribution, and different integration rules are some of the methods used to find these matrices. In this finite element program, integration using Gauss-Quadrature (see Section 3.5) points is used to determine the element stiffness matrices. This method is a type of numerical integration scheme that approximates the integral expression through the use of algebraic expressions.

Displacements $\{u\}$ are interpolated over an element in the same way as done for the field quantity ϕ as

$$\{u\} = [N]\{d\} \quad \text{where} \quad \{u\} = [u \ v \ w]^T \quad (3-6)$$

The vector $\{d\}$ is the nodal displacements of the DOF for the element. From the strain-displacement relations we have

$$\epsilon_x = \frac{\partial u}{\partial x} \quad \epsilon_y = \frac{\partial v}{\partial y} \quad \gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \quad (3-7)$$

Therefore, in general form, strain displacement relations can be written as

$$\{\epsilon\} = [\partial] \{u\} \quad (3-8)$$

Substituting equation 3-6 in 3-8 we get

$$\{\epsilon\} = [B] \{d\} \quad \text{where} \quad [B] = [\partial][N] \quad (3-9)$$

The matrix $[B]$ is called the strain-displacement matrix.

For a 2-D plain stress element having isotropic material properties, the constitutive matrix is a symmetrical 3x3 matrix given by

$$[E] = \frac{E}{(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{(1-\nu)}{2} \end{bmatrix} \quad (3-10)$$

where E is the Modulus of Elasticity of the material and ν is Poisson's ratio.

Finally the element stiffness matrix can be formulated as

$$[K] = \int [B]^T [E] [B] dV \quad (3-11)$$

where $[B]$ and $[E]$ are obtained from equations 3.9 and 3.10, respectively. In the case of 2-D analysis this equation can be written as

$$[K] = \iint [B]^T [E] [B] t \, dx \, dy \quad (3-12)$$

where t is the thickness of the element usually assumed to be constant throughout the element and taken as unity.

3.4 Element Boundary and Loading Conditions

Before the solution of the structural equations, both boundary conditions and loading conditions must be prescribed for the system. Without boundary conditions the structural equations will not produce a single unique solution for the prescribed loading conditions. As such, the structural system will have rigid body motions. Without loading conditions the structural equations will produce no displacements of the structural system. Thus, it is necessary that both boundary and loading conditions be prescribed for the structural system. These conditions are prescribed at particular DOF of the structural system.

Boundary conditions, or support conditions, include prescriptions of displacements on the sides or surfaces of a structural system. The boundary conditions for the global element are set by stiffening the DOF for the corresponding local element or forcing the structural equations to be equal to a unique displacement at that DOF. These boundary conditions can be translational or rotational depending on the DOF that needs to be modified. In this finite element program, only translational DOF are considered in the finite elements (i.e., there are no rotational DOF) and therefore only translational displacements are restricted with respect to particular support conditions such as a fixed-support, pinned-support, or roller-support.

Loading conditions are prescribed in a fashion similar to boundary/support conditions. Depending on the location of the prescribed load, the DOF corresponding to that location and direction are modified. In general, a distributed load can act normally or tangentially

to a boundary of the element. These loads can be expressed as surface tractions, which are forces per unit of surface area, directed parallel to the coordinate axis. Concentrated forces are prescribed directly to the DOF of the structural system.

3.5 Solution Techniques

Each element contributes to the overall stiffness of the structural system. These elements must be assembled to construct the structural system. This process is known as the assembly process. DOF for each element are assigned global DOF values. The stiffness corresponding to each DOF are then assembled or “fed” into the global stiffness of the system. It is at this point that the boundary conditions and loading conditions are truly applied to the system. After application of the boundary and loading conditions, solution of the structural equations may commence.

There are a variety of techniques to solve the system of simultaneous equations that are assembled in the assembly process. In this finite element analysis program, the back-substitution method of Gauss-elimination is utilized to solve the structural equations. In Gauss-elimination, the structural equations $[K]\{D\} = \{R\}$ are solved for the system displacements $\{D\}$ by reducing the system stiffness $[K]$ to upper triangular form. The system displacements are then determined in reverse order through back substitution.

3.6 Advanced Finite Element Formulations

As stated previously, the primary objective of this study is to incorporate advanced finite elements into a finite element analysis program. Sections 3.6.1 through 3.6.5 provide the theoretical details pertaining to these elements. The advanced elements used herein are the Constant Strain Triangular element (CST), the Linear Strain Triangular element (LST), the Quadratic Four-Node element (Q4), and the Quadratic Eight-Node element (Q8). The following sections also provide the shape functions, strain-displacement matrices and the formulation of the element stiffness matrix for each element. In addition, the procedure for these elements in natural coordinates as isoparametric elements is also discussed.

3.6.1 Isoparametric Elements

Isoparametric elements are used as a versatile formation in this finite element analysis program which helps in easier numerical integration of the elements using Gauss quadrature. Shape functions are based on the natural coordinates rather than the original coordinates. In this section, the shape functions and element stiffness matrix are illustrated for a Q4 element.

To explain the concept of isoparametric elements, consider a rectangular Q4 element as shown in Figure 3-1. Isoparametric elements use auxiliary coordinates or reference coordinates termed as ξ and η referred to as the natural coordinates. During the

formulation of the element stiffness, the coordinate transformation from the natural coordinates to the actual or physical (x and y) coordinates is taken care of through the use of a transformation function that maps the two coordinate systems. For these elements, shape functions are used to interpolate both the displacement field and element geometry. For the Q4 isoparametric element, the individual shape functions are

$$\begin{aligned}
 N_1 &= \frac{1}{4}(1 - \xi)(1 - \eta) \\
 N_2 &= \frac{1}{4}(1 + \xi)(1 - \eta) \\
 N_3 &= \frac{1}{4}(1 + \xi)(1 + \eta) \\
 N_4 &= \frac{1}{4}(1 - \xi)(1 + \eta)
 \end{aligned} \tag{3-13}$$

These shape functions may also be obtained by considering the shape functions for the physical coordinate system given in a Section 3.6.4 (see equation 3-31). Setting $a = 1$, $b = 1$, $x = \xi$ and $y = \eta$ the shape functions in equation 3-13 are again obtained.

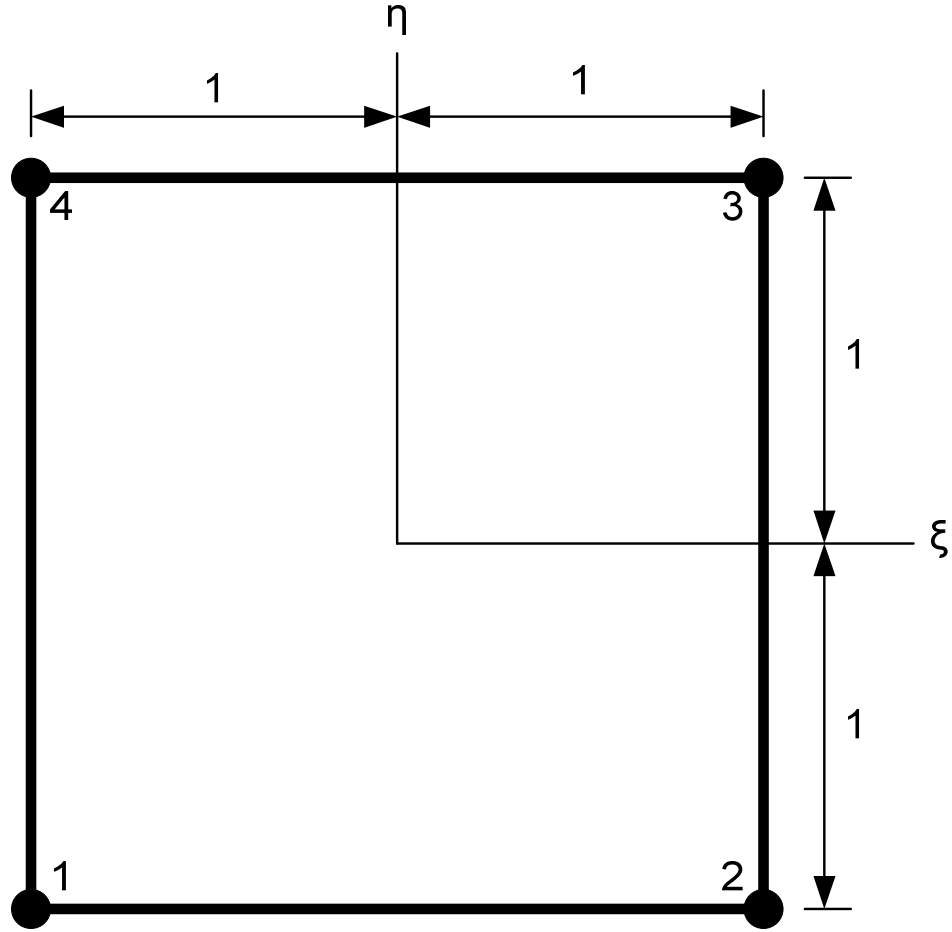


Figure 3-1: Four Node Isoparametric Element

The mapping from the natural coordinate system to the physical coordinate system is accomplished through the use of a transformation function as previously described. To accomplish the mapping, the Jacobian is utilized which is simply a scale factor that multiplies $d\xi d\eta$ to produce the physical area increment $dx dy$. The Jacobian matrix for the Q4 isoparametric element is given by

$$[J] = \frac{1}{4} \begin{bmatrix} -(1-\eta) & (1-\eta) & (1+\eta) & -(1+\eta) \\ -(1-\xi) & -(1+\xi) & (1+\xi) & (1-\xi) \end{bmatrix} \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_4 \end{bmatrix} = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix} \quad (3-14)$$

The strain-displacement matrix, $[B]$, is determined according to equation 3-9. However, derivatives are now made with respect to the natural coordinate system rather than the physical coordinate system. Finally, the element stiffness matrix is determined in a fashion similar to equation 3-12. Thus, for an isoparametric element the element stiffness matrix is determined as

$$[K] = \iint [B]^T [E] [B] t \, dx \, dy = \int_{-1}^1 \int_{-1}^1 [B]^T [E] [B] t J \, d\xi \, d\eta \quad (3-15)$$

where J is the determinant of the Jacobian determined by equation 3-14. To accomplish the integration as given in equation 3-15, from a numerical perspective, Gaussian-Quadrature is used. The Gauss product rule is obtained by successive application of a one-dimensional Gauss rule. For the function $\phi = \phi(\xi, \eta)$, the Quadrature rule is given as

$$I = \int_{-1}^1 \int_{-1}^1 \phi(\xi, \eta) d\xi d\eta = \sum_i \sum_j W_i W_j \phi(\xi, \eta) \quad (3-16)$$

where W_i and W_j are scalar weight functions corresponding to the gauss points i and j .

3.6.2 Constant Strain Triangular Elements (CST)

Linear triangles are plane triangles whose field quantity varies linearly with coordinates x and y . The linear triangular element is called a constant strain triangular (CST) element as the displacement fields produce a constant strain field.

As shown in Figure 3-2, the coordinates of node 1 are at $x = y = 0$ and side 1-2 is along the local x -axis. From the principles of interpolation, the field quantity ϕ is interpolated over the element by the polynomial in terms of generalized DOF a_i in the u and v directions as

$$u = [1 \ x \ y] \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \end{Bmatrix} \quad (3-17)$$

$$v = [1 \ x \ y] \begin{Bmatrix} a_4 \\ a_5 \\ a_6 \end{Bmatrix} \quad (3-18)$$

Applying the coordinates from Figure 3-2 for each of the nodes to equation 3-17 and 3-18, then equation 3-3 produces

$$\begin{Bmatrix} \emptyset_1 \\ \emptyset_2 \\ \emptyset_3 \end{Bmatrix} = [A] \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \end{Bmatrix} \quad \text{where} \quad [A] = \begin{bmatrix} 1 & 0 & 0 \\ 1 & x_2 & 0 \\ 1 & x_3 & y_3 \end{bmatrix} \quad (3-19)$$

$$\begin{Bmatrix} \emptyset_4 \\ \emptyset_5 \\ \emptyset_6 \end{Bmatrix} = [A] \begin{Bmatrix} a_4 \\ a_5 \\ a_6 \end{Bmatrix} \quad \text{where} \quad [A] = \begin{bmatrix} 1 & 0 & 0 \\ 1 & x_2 & 0 \\ 1 & x_3 & y_3 \end{bmatrix} \quad (3-20)$$

Using these relationships, the shape functions in the physical coordinate system can be determined from equation 3-4 as

$$\begin{aligned} N_1 &= 1 - \frac{1}{x_2} * x + \frac{(x_3 - x_2)}{x_2 y_3} * y \\ N_2 &= \frac{1}{x_2} * x - \frac{x_3}{x_2 y_3} * y \\ N_3 &= \frac{1}{y_3} * y \end{aligned} \quad (3-21)$$

The strain-displacement matrix $[B]$ is calculated from equation 3-9 as

$$[B] = \begin{bmatrix} \frac{-1}{x_2} & 0 & \frac{1}{x_2} & 0 & 0 & 0 \\ 0 & \frac{x_3 - x_2}{x_2 y_3} & 0 & \frac{-x_3}{x_2 y_3} & 0 & \frac{1}{y_3} \\ \frac{x_3 - x_2}{x_2 y_3} & \frac{-1}{x_2} & \frac{-x_3}{x_2 y_3} & \frac{1}{x_2} & \frac{1}{y_3} & 0 \end{bmatrix} \quad (3-22)$$

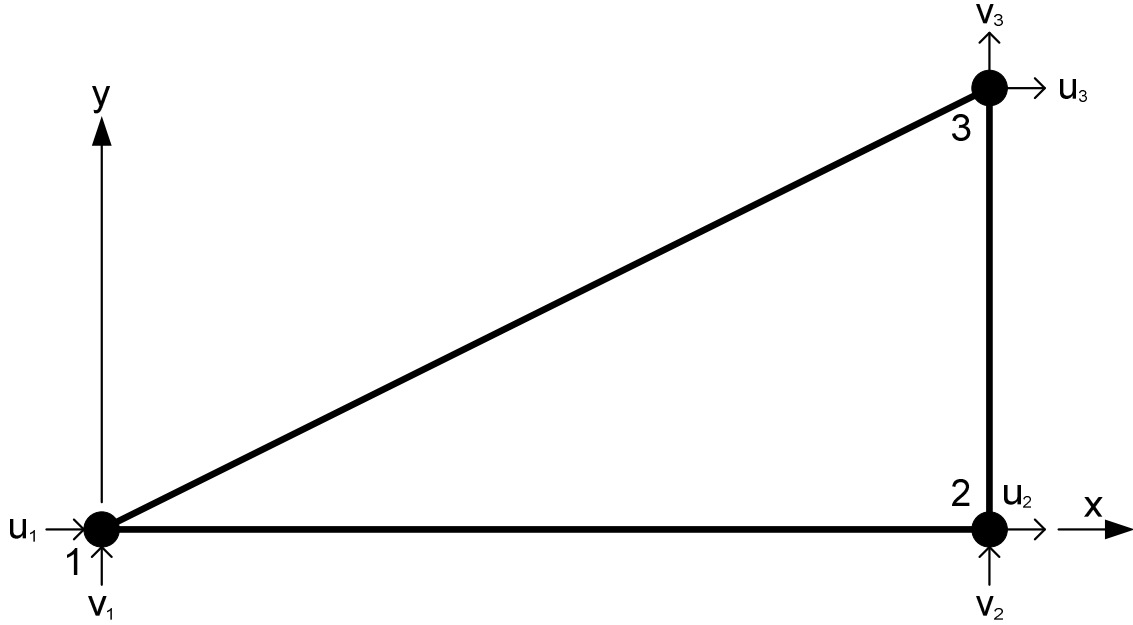


Figure 3-2: Three Node CST Element

As the strain-displacement matrix, element thickness (assumed to be unity), and the constitutive matrix $[E]$ are constants (independent of x and y), equation 3-12 used in determining the 6x6 element stiffness matrix for the CST element simplifies to

$$[K] = [B]^T [E] [B] A \quad (3-23)$$

where A is the area of the triangle.

The constant strain triangular element was the first element devised for plane stress analysis. It does not work very well. In bending, a mesh of these elements is undesirably stiff. Correct results can be approached with the refinement of the mesh, but the convergence is slow. In plane strain conditions, a mesh can lock so that it cannot deform at all.

The CST model predicts y direction displacement and σ_x stresses that are only one-quarter the correct value. The inability of the CST to represent linearly varying stress and strain is partly to blame for this poor result. The CST element also displays a spurious shear stress. It also displays a transverse shear strain, which should not be present. The spurious shear strain absorbs energy, so that if a given deformation is prescribed, the load needed to produce it is larger than the correct value. This is the reason why there is excessive stiffness in bending of the CST element.

The CST element can also be represented in an isoparametric element form. The shape functions for the CST isoparametric element are given as

$$\begin{aligned} N_1 &= 1 - \xi - \eta \\ N_2 &= \xi \\ N_3 &= \eta \end{aligned} \tag{3-24}$$

In the development of the FEA program written using Matlab, the isoparametric shape functions are used in the element formulation. Verification of these formulations is provided in Appendix B of this study.

3.6.3 Linear Strain Triangular Elements (LST)

Similar to the CST element, the linear strain triangular element (LST) are quadratic triangles whose field quantity varies quadratically with coordinates x and y . The quadratic triangular element is called a linear strain triangular (LST) element as the displacements fields produce a linear variation in the strain field.

As shown in Figure 3-3, the triangular element has mid-side nodes in addition to the vertex nodes. For displacement-based analysis, the nodal DOF are u_i and v_i for each node $i = 1, 2, \dots, 6$ for a total of 12 DOF per element. For the generalized DOF a_i , the element displacement is interpolated by the polynomial as

$$u = [1 \quad x \quad y \quad x^2 \quad xy \quad y^2] \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{Bmatrix} \quad (3-25)$$

$$v = [1 \quad x \quad y \quad x^2 \quad xy \quad y^2] \begin{Bmatrix} a_7 \\ a_8 \\ a_9 \\ a_{10} \\ a_{11} \\ a_{12} \end{Bmatrix} \quad (3-26)$$

In a similar fashion to the CST element, the shape functions are found as

$$\begin{aligned} N_1 &= 1 - \frac{3x}{a} - \frac{3y}{b} + \frac{2x^2}{a^2} + \frac{2xy}{ab} + \frac{2y^2}{b^2} \\ N_2 &= \frac{-x}{a^2} + \frac{2x^2}{a^2} + \frac{2xy}{ab} \\ N_3 &= \frac{y}{b} + \frac{2y^2}{b^2} \\ N_4 &= \frac{4x}{a} - \frac{4x^2}{a^2} \\ N_5 &= \frac{2xy}{ab} \\ N_6 &= \frac{-4y}{b} - \frac{2xy}{ab} + \frac{4y^2}{b^2} \end{aligned} \quad (3-27)$$

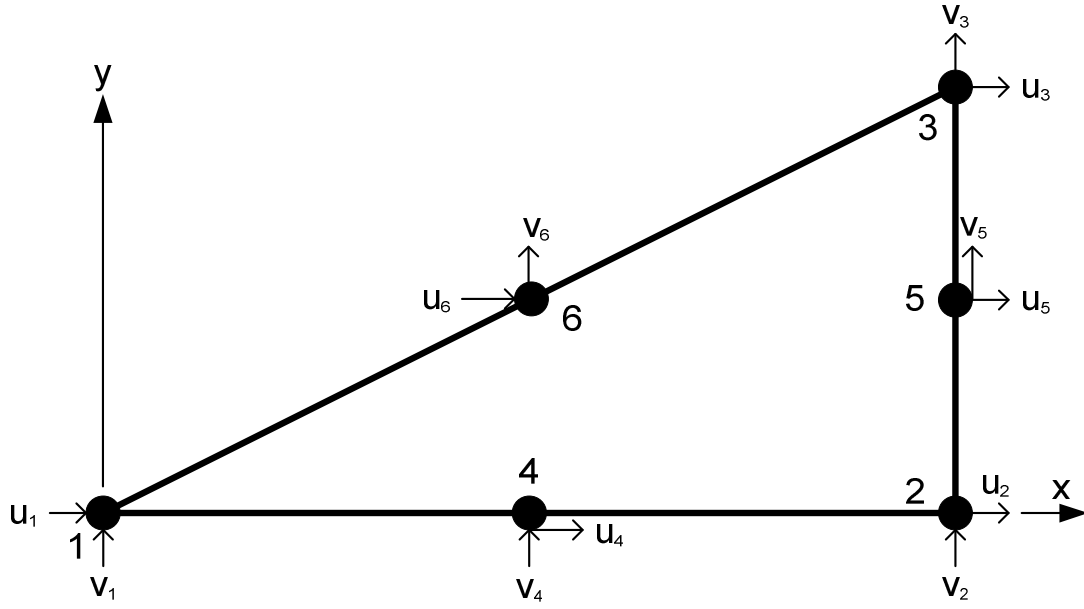


Figure 3-3: Six Node LST Element

The strain-displacement matrix is determined from equation 3-9. The resulting 12x12 element stiffness matrix for the LST element is determined from equation 3-12 without any simplifications as was done for the CST element. As such, it is advantageous to use the isoparametric form of the LST element in a numerical integration scheme to determine the element stiffness matrix. The shape functions for the LST isoparametric element are given as

$$\begin{aligned}
 N_1 &= 2(1 - \xi - \eta)\left(\frac{1}{2} - \xi - \eta\right) \\
 N_2 &= 2\xi\left(\xi - \frac{1}{2}\right) \\
 N_3 &= 2\eta\left(\eta - \frac{1}{2}\right) \\
 N_4 &= 4\xi(1 - \xi - \eta) \\
 N_5 &= 4\xi\eta \\
 N_6 &= 4\eta(1 - \xi - \eta)
 \end{aligned} \tag{3-28}$$

In the development of the FEA program written using Matlab, the isoparametric shape functions are used in the element formulation. Verification of these formulations is provided in Appendix B of this study.

3.6.4 Bilinear Rectangular Elements (Q4)

A bilinear rectangular element (Q4), similar to the CST elements, has a field quantity that also varies linearly with coordinates x and y . As shown in Figure 3-4, the Q4 element has eight total DOF with two DOF at each of the four nodes. The displacements are also interpolated over the element by the polynomial in terms of generalized DOF a_i in the u and v directions as

$$u = [1 \quad x \quad y \quad xy] \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{Bmatrix} \quad (3-29)$$

$$v = [1 \quad x \quad y \quad xy] \begin{Bmatrix} a_5 \\ a_6 \\ a_7 \\ a_8 \end{Bmatrix} \quad (3-30)$$

In a similar fashion to the CST element, the shape functions are found as

$$\begin{aligned} N_1 &= \frac{1}{4ab}(1-x)(1-y) \\ N_2 &= \frac{1}{4ab}(1+x)(1-y) \\ N_3 &= \frac{1}{4ab}(1+x)(1+y) \\ N_4 &= \frac{1}{4ab}(1-x)(1+y) \end{aligned} \quad (3-31)$$

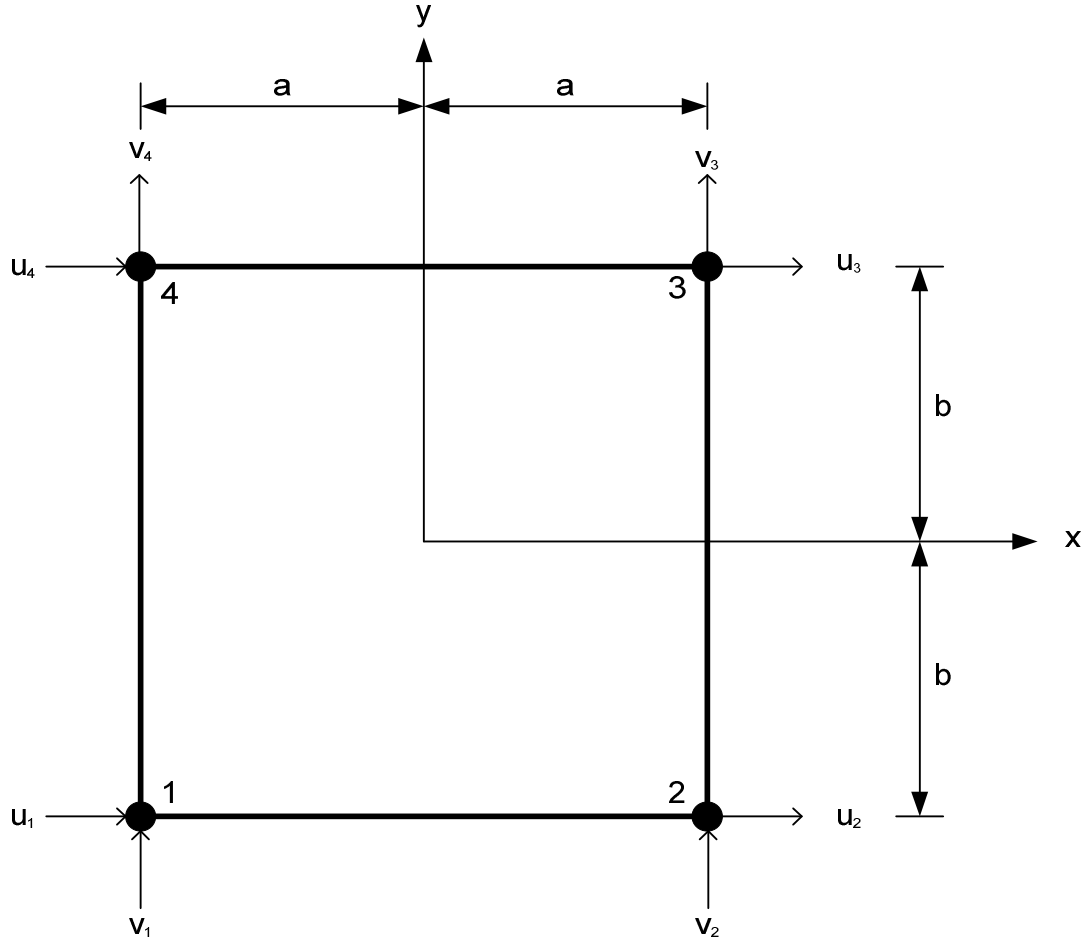


Figure 3-4: Four Node Q4 Element

Using the shape functions from equation 3-31 and substituting them in equation 3-9 gives the strain displacement matrix as

$$[B] = \frac{1}{4ab} \begin{bmatrix} -(b-y) & 0 & (b-y) & 0 & (b+y) & 0 & -(b+y) & 0 \\ 0 & -(a-x) & 0 & -(a+x) & 0 & (a+x) & 0 & (a-x) \\ -(a-x) & -(b-y) & -(a+x) & (b-y) & (a+x) & (b+y) & (a-x) & (b+y) \end{bmatrix} \quad (3-32)$$

Similar to equation 3-12, the 8x8 element stiffness matrix is determined using the strain-displacement matrix as

$$[K] = \int_{-b}^b \int_{-a}^a [B]^T [E] [B] t \, dx \, dy \quad (3-33)$$

Like the CST, the Q4 element cannot exhibit pure bending. When bent, it displays shear strain as well as the expected bending strain. The parasitic shear absorbs strain energy, so that if a given bending deformation is prescribed, the bending moment needed to produce it is larger than the correct value. In other words, Q4 element exhibits shear locking behavior.

As for the CST and LST elements, the isoparametric shape functions are given as

$$\begin{aligned} N_1 &= \frac{1}{4}(1 - \xi)(1 - \eta) \\ N_2 &= \frac{1}{4}(1 + \xi)(1 - \eta) \\ N_3 &= \frac{1}{4}(1 + \xi)(1 + \eta) \\ N_4 &= \frac{1}{4}(1 - \xi)(1 + \eta) \end{aligned} \tag{3-34}$$

In the development of the FEA program written using Matlab, the isoparametric shape functions are used in the element formulation. Verification of these formulations is provided in Appendix B of this study.

3.6.5 Quadratic Rectangular Element (Q8)

The quadratic rectangular element (Q8), analogous to the LST element as compared to the CST element, is different from the Q4 element in that the field quantity varies quadratically with coordinates x and y . A quadratic Q8 rectangular element is shown in Figure 3-5. The element has sixteen DOF with mid-side nodes between the corner nodes of the Q4 element. In terms of generalized DOF a_i , the displacement field of the element is given by

$$u = [1 \quad x \quad y \quad x^2 \quad xy \quad y^2 \quad x^2y \quad xy^2] \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{Bmatrix} \quad (3-35)$$

$$v = [1 \quad x \quad y \quad x^2 \quad xy \quad y^2 \quad x^2y \quad xy^2] \begin{Bmatrix} a_9 \\ a_{10} \\ a_{11} \\ a_{12} \\ a_{13} \\ a_{14} \\ a_{15} \\ a_{16} \end{Bmatrix} \quad (3-36)$$

In a similar fashion to the CST element, the shape functions are found as

$$\begin{aligned} N_1 &= \frac{-1}{4} + \frac{x^2}{4a^2} + \frac{xy}{4ab} + \frac{y^2}{4b^2} - \frac{x^2y}{4a^2b} - \frac{xy^2}{4ab^2} \\ N_2 &= \frac{1}{2} - \frac{x}{2a} - \frac{y^2}{2b^2} + \frac{xy^2}{2ab^2} \\ N_3 &= \frac{-1}{4} + \frac{x^2}{4a^2} - \frac{xy}{4ab} + \frac{y^2}{4b^2} + \frac{x^2y}{4a^2b} - \frac{xy^2}{4ab^2} \\ N_4 &= \frac{1}{2} - \frac{y}{2b} - \frac{x^2}{2a^2} + \frac{x^2y}{2a^2b} \\ N_5 &= \frac{1}{2} + \frac{y}{2b} - \frac{x^2}{2a^2} - \frac{x^2y}{2a^2b} \\ N_6 &= \frac{-1}{4} + \frac{x^2}{4a^2} - \frac{xy}{4ab} + \frac{y^2}{4b^2} - \frac{x^2y}{4a^2b} + \frac{xy^2}{4ab^2} \\ N_7 &= \frac{1}{2} + \frac{x}{2a} - \frac{y^2}{2b^2} - \frac{xy^2}{2ab^2} \\ N_8 &= \frac{-1}{4} + \frac{x^2}{4a^2} + \frac{xy}{4ab} + \frac{y^2}{4b^2} + \frac{x^2y}{4a^2b} + \frac{xy^2}{4ab^2} \end{aligned} \quad (3-37)$$

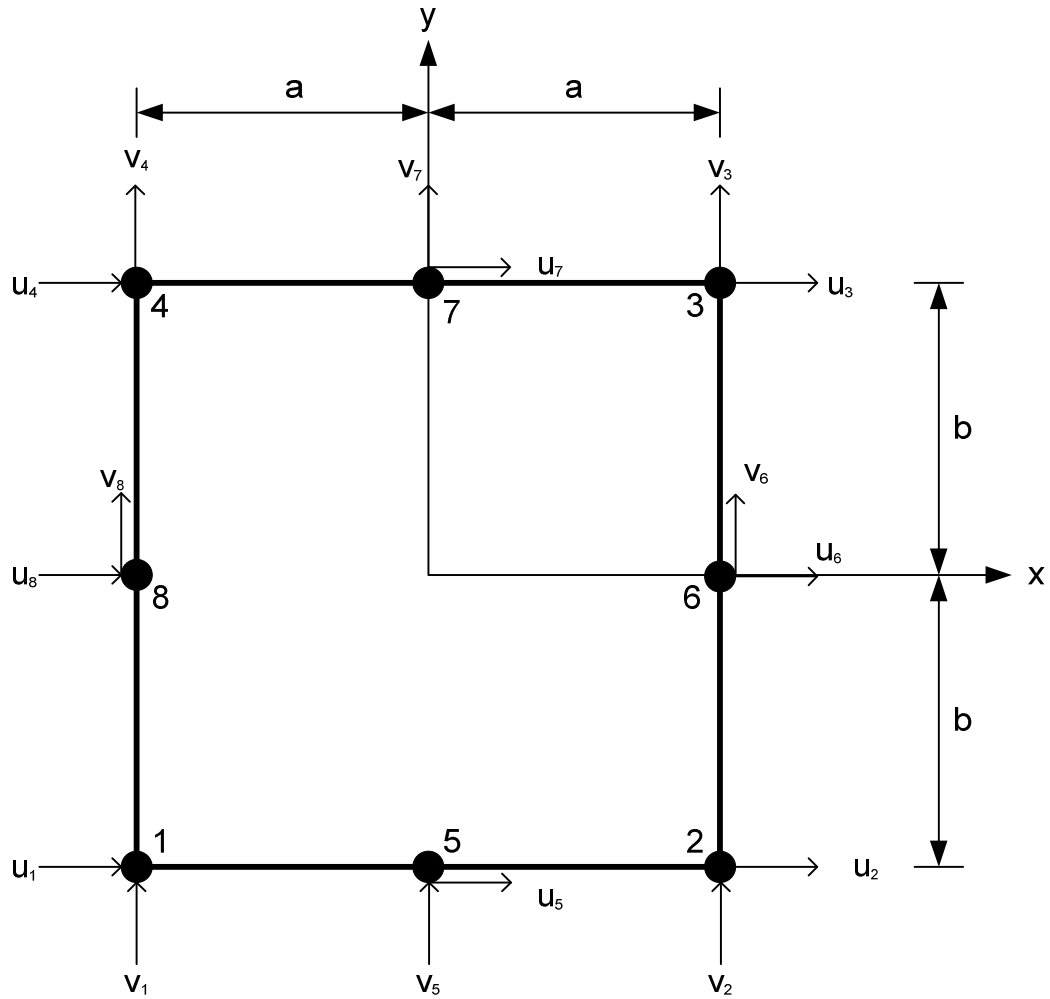


Figure 3-5: Eight Node Q8 Element

The strain-displacement matrix is determined from equation 3-9. The resulting element stiffness is determined from equation 3-12 without any simplifications. As such, it is advantageous to use the isoparametric form of the Q8 element in a numerical integration scheme to determine the element stiffness matrix. The shape functions for the Q8 isoparametric element are given as

$$\begin{aligned}
N_1 &= \frac{1}{4}(1 - \xi)(1 - \eta)(-\xi - \eta - 1) \\
N_2 &= \frac{1}{4}(1 + \xi)(1 - \eta)(\xi - \eta - 1) \\
N_3 &= \frac{1}{4}(1 + \xi)(1 + \eta)(\xi + \eta - 1) \\
N_4 &= \frac{1}{4}(1 - \xi)(1 + \eta)(-\xi + \eta - 1) \\
N_5 &= \frac{1}{2}(1 - \xi^2)(1 - \eta) \\
N_6 &= \frac{1}{2}(1 - \eta^2)(1 + \xi) \\
N_7 &= \frac{1}{2}(1 - \xi^2)(1 + \eta) \\
N_8 &= \frac{1}{2}(1 - \eta^2)(1 - \xi)
\end{aligned} \tag{3-38}$$

In the development of the FEA program written using Matlab, the isoparametric shape functions are used in the element formulation. Verification of these formulations is provided in Appendix B of this study.

CHAPTER IV

CASE STUDIES

4.1 Introduction

Case studies are provided only to illustrate the capabilities of the Matlab FEA program. The specific accuracy of the Matlab FEA program is not of concern at this time because a specific problem was not considered in this study. The main program is provided in Appendix A along with the general input file for the program. Sub-routines of the FEA program have not been provided with this current study as the program is still under substantial development at Oklahoma State University.

The case studies are provided for each of the four elements discussed in Chapter III of this study. The illustrative example for each element consists of analyzing the same problem. This enables for direct comparisons between the four types of elements considered in this study. Again, these comparisons are only made with respect to the capabilities of the Matlab FEA program and not the accuracy of any given analysis. The details of the problem are presented in this first section followed by the results from each element type in Sections 4.2 – 4.5.

A cantilever beam fixed at one end and statically loaded at the other end, as shown in Figure 4-1, is used to analyze the behavior of the various elements. Only a single concentrated load of 100 lb is applied to the free end of the beam. The length of the beam is 15ft (180in) and its depth is 1ft (12in). All analyses are considered to be a 2-D analysis, thus the thickness (or width) of the beam is assumed to be unity. The modulus of elasticity (E) of the beam is taken as 10^6 lb/in² and Poisson's Ratio as 0.3.

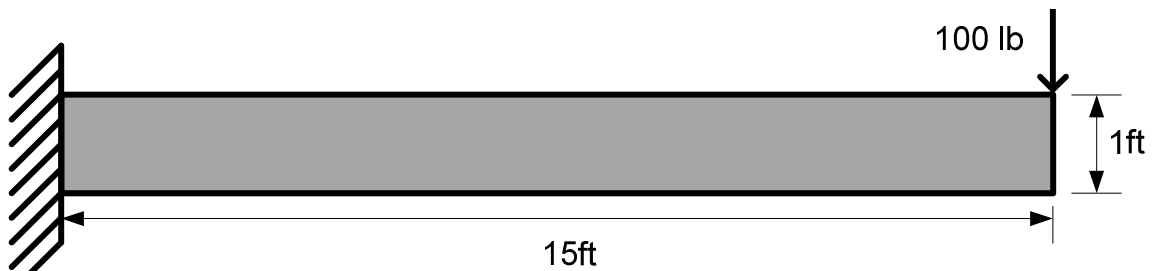


Figure 4-1: Fixed Beam Loaded at the Free End

For each analysis given in the following sections, the general finite element input values are outlined in Table 4-1. An adequate number of elements have been chosen in each case listed in Table 4-1. The CST and Q4 element case studies include a slight refinement of the mesh to provide the reader with an understanding of the importance of the finite element mesh. Further refinement of the mesh could be implemented, but the purpose of this study was to only implement and confirm the use of a computer program written in Matlab for performing finite element analysis using advanced finite element formulations.

Element Type	Beam Length (in)	Beam Width (in)	Number of Elements	Number of Nodes	Modulus of Elasticity (psi)	Poisson's Ratio
CST	180	12	20 / 40	22	10^6	0.3
LST	180	12	20	63	10^6	0.3
Q4	180	12	10 / 20	22	10^6	0.3
Q8	180	12	10	53	10^6	0.3

Table 4-1: Input Values for Various Elements

4.2 CST Elements

The beam of Figure 4-1 is modeled using 20 and 40 CST elements. Figure 4-2 provides the finite element mesh generated by the Matlab program written as a result of this study. Sub-figure (a) is provided with a series of 20 CST elements only along the length of the beam. Due to the limitations of this element, this type of finite element mesh will not accurately predict the response of the beam. As such, sub-figure (b) provides an alternative mesh, using twice the number of total elements (40 elements), that will more accurately model the behavior of the beam. The reason for this alternative mesh is the inability of the CST element to model pure bending accurately. Each node is also labeled in Figure 4-2. The DOF at the fixed end of the beam are constrained to zero displacement. The loading is applied to the DOF located at the top right of the mesh at the tip of the free end of the beam. The results of the analysis are shown in Figures 4-3

through 4-5. Figure 4-3 shows the displacement of the CST elements when a concentrated load is applied at the free end of a fixed beam. The contour plots for the deflections are shown in Figures 4-4 and 4-5 for the x and y directions, respectively. For each figure, Figures 4-3 through 4-5, each finite element mesh is also represented similar to Figure 4-2.

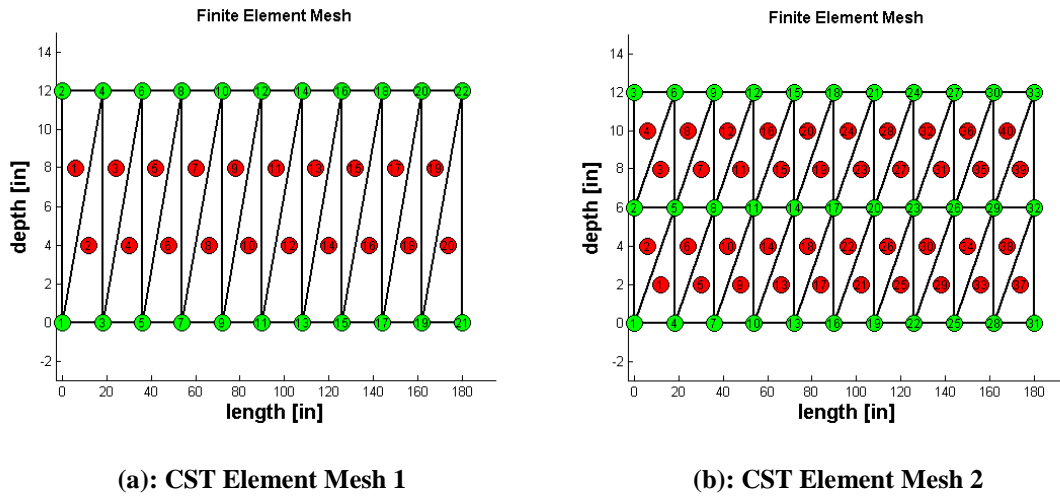


Figure 4-2: CST Element Mesh

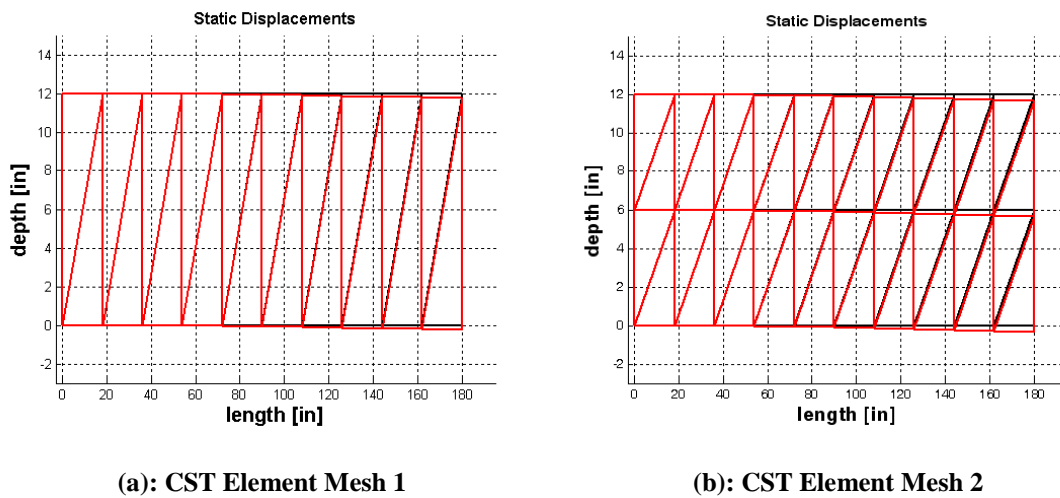
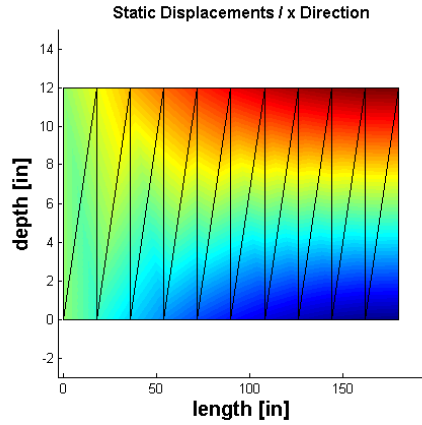
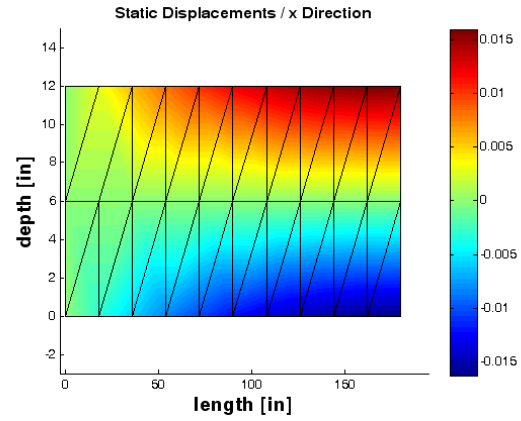


Figure 4-3: CST Beam Deflection Profile

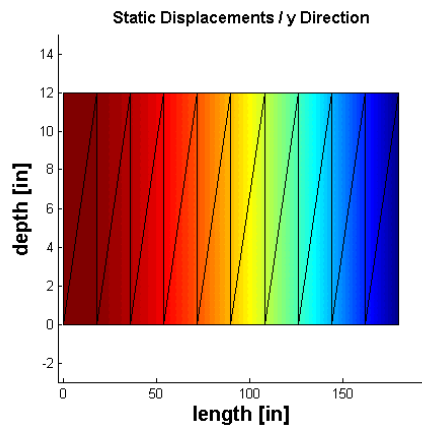


(a): CST Element Mesh 1

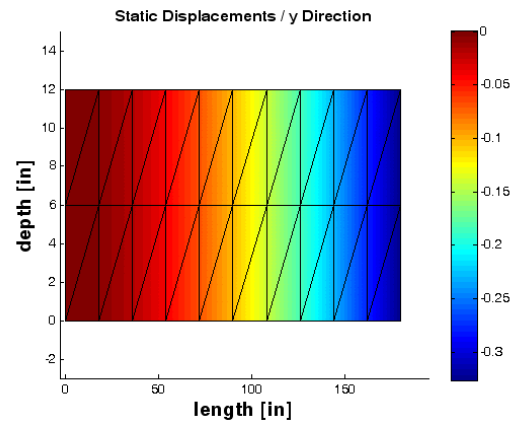


(b): CST Element Mesh 2

Figure 4-4: Contour Plot for CST Element Displacements in the x-Direction



(a): CST Element Mesh 1



(b): CST Element Mesh 2

Figure 4-5: Contour Plot for CST Element Displacements in the y-Direction

Table 4-2 shows the maximum tip deflection to be 0.2328 inches in the negative y-direction for the first finite element mesh of 20 elements provided in sub-figures (a). Table 4-2 also provides the results from the alternative finite element mesh of 40 elements provided in sub-figures (b). From these results, and the comparison to an analytical solution to be provided later, it is noted that the CST elements are undesirably stiff and, thus, the deflection of the beam in y-direction is minimal.

Finite Element Mesh	Tip Displacement (in)
20 Elements <i>CST Element Mesh 1</i>	-0.2328
40 Elements <i>CST Element Mesh 2</i>	-0.3262

Table 4-2: Maximum Tip Deflection for CST Elements

Due to the linear interpolation used during the formulation of the shape functions, the deflection of the nodes is minimal. This type of the element does not work well due to the lower number of DOF per element. Further refinement of the finite element mesh will help in convergence to the exact value but is not considered in this study.

4.3 LST Elements

The beam of Figure 4-1 is modeled using 20 LST elements. Figure 4-6 provides the finite element mesh generated by the Matlab program written as a result of this study. Each node is also labeled in Figure 4-6. DOF 1 – 6, located at nodes 1, 2 and 3, at the fixed end of the beam are constrained to zero displacement. The loading is applied to DOF 126, located at node 63, at the tip of the free end of the beam. The results of the analysis are shown in Figures 4-7 through 4-9. Figure 4-7 shows the displacement of the LST elements when a concentrated load is applied at the free end of a fixed beam. The

contour plots for the deflections are shown in Figures 4-8 and 4-9 for the x and y directions, respectively.

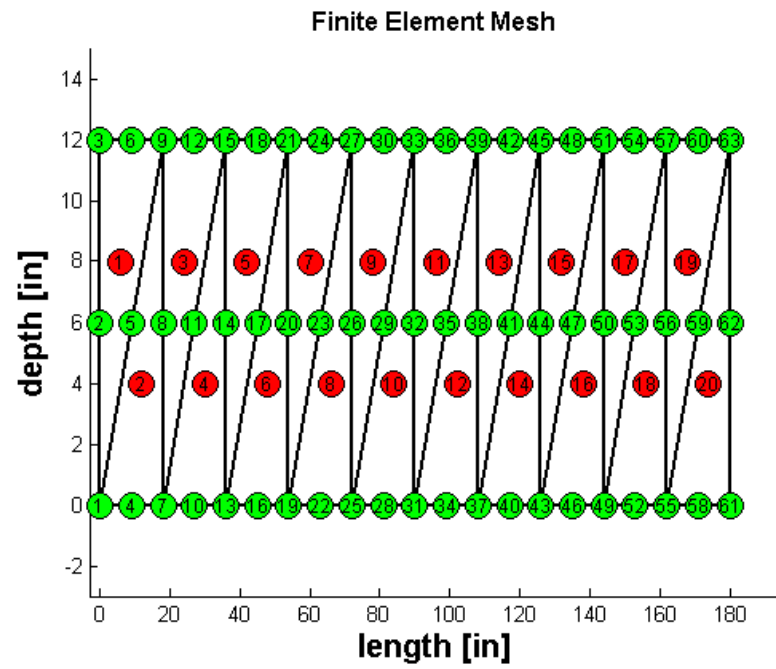


Figure 4-6: LST Element Mesh

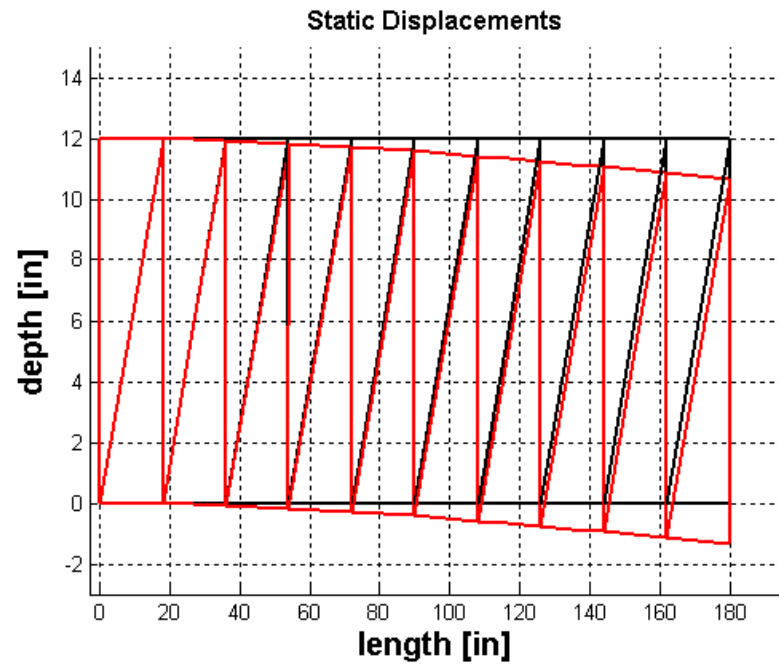


Figure 4-7: LST Beam Deflection Profile

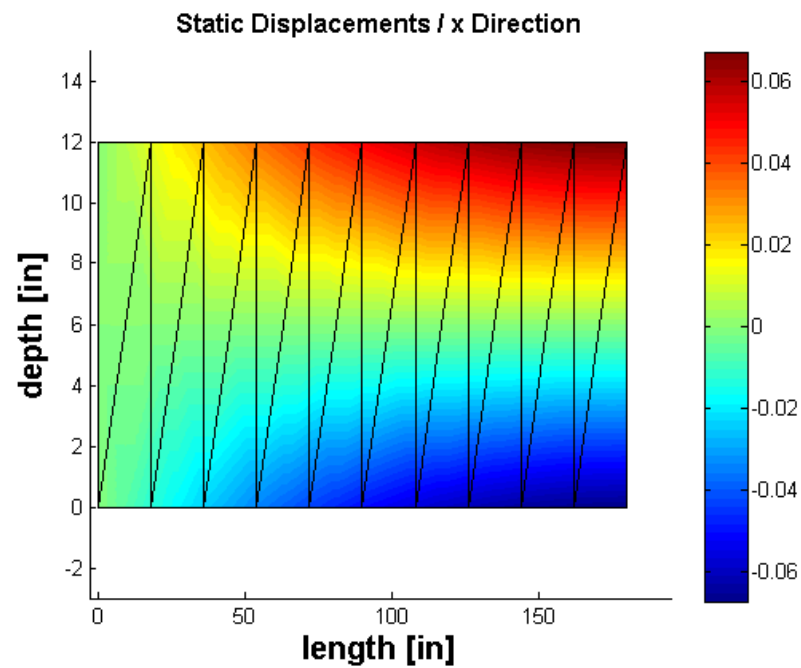


Figure 4-8: Contour Plot for LST Element Displacements in the x-Direction

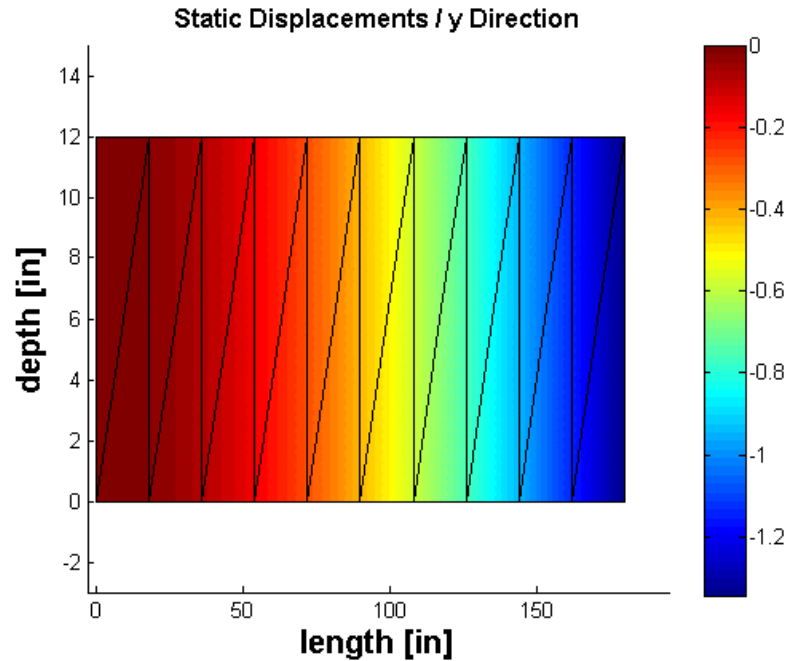


Figure 4-9: Contour Plot for LST Element Displacements in the y-Direction

Figure 4-9 shows the displacement of the LST elements when a concentrated load is applied at the free end of the beam. Figure 4-8 and 4-9 show a contour plot with a smooth displacement of the nodes in both the x and y directions. This is due to the quadratic interpolation used to formulate the element stiffness matrices. The displacement in x direction is positive to negative from the top to the bottom nodes respectively. Table 4-3 shows a maximum tip deflection of -1.3433in in the y -direction.

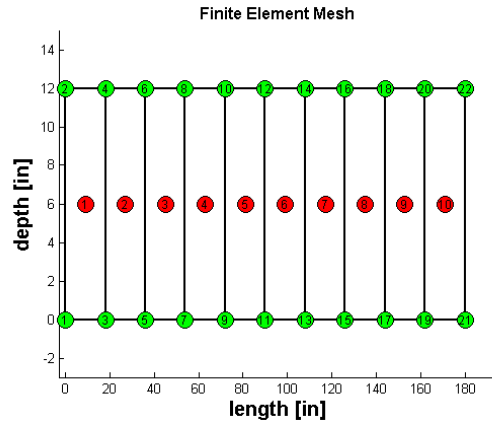
Finite Element Mesh	Tip Displacement (in)
20 Elements <i>LST Element Mesh</i>	-1.3433

Table 4-3: Maximum Tip Deflection for LST Elements

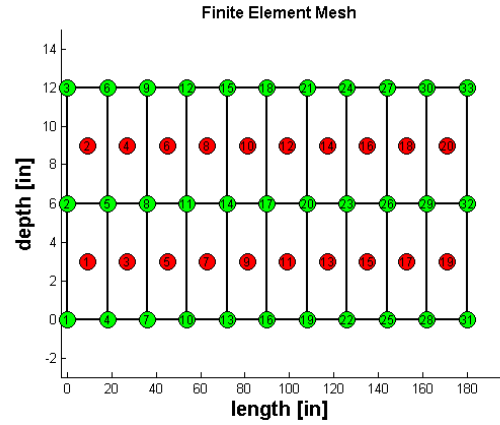
Due to the quadratic interpolation used to formulate the shape functions, the deflections of the nodes are well refined. In modeling this particular problem, the only shortcoming of the LST elements is that γ_{xy} is represented as linear rather than quadratic in y .

4.4 Q4 Elements

The beam of Figure 4-1 is modeled using 10 and 20 Q4 elements. Figure 4-10 provides the finite element mesh generated by the Matlab program written as a result of this study. Sub-figure (a) is provided with a series of 10 Q4 elements only along the length of the beam. Due to the limitations of this element, this type of finite element mesh will not accurately predict the response of the beam. As such, sub-figure (b) provides an alternative mesh, using twice the number of total elements (20 elements), that will more accurately model the behavior of the beam. The reason for this alternative mesh is the inability of the Q4 element to model pure bending accurately. Each node is also labeled in Figure 4-10. The DOF at the fixed end of the beam are constrained to zero displacement. The loading is applied to the DOF located at the top right of the mesh at the tip of the free end of the beam. The results of the analysis are shown in Figures 4-11 through 4-13. Figure 4-11 shows the displacement of the Q4 elements when a concentrated load is applied at the free end of a fixed beam. The contour plots for the deflections are shown in Figures 4-12 and 4-13 for the x and y directions, respectively. For each figure, Figures 4-11 through 4-13, each finite element mesh is also represented similar to Figure 4-10.

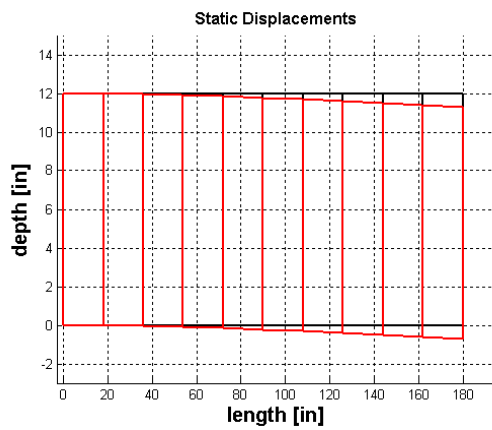


(a): Q4 Element Mesh 1

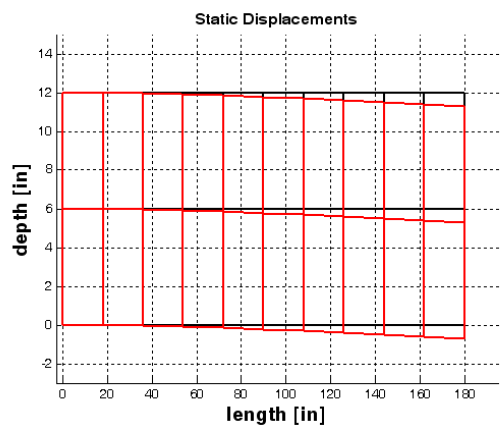


(b): Q4 Element Mesh 2

Figure 4-10: Q4 Element Mesh

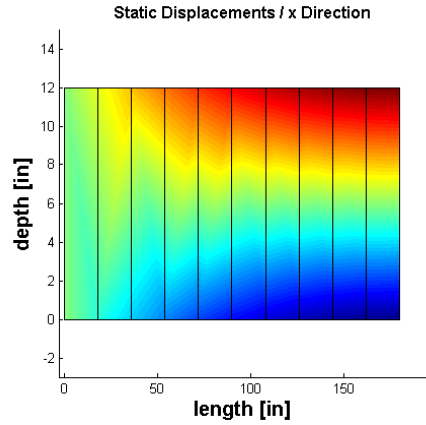


(a): Q4 Element Mesh 1

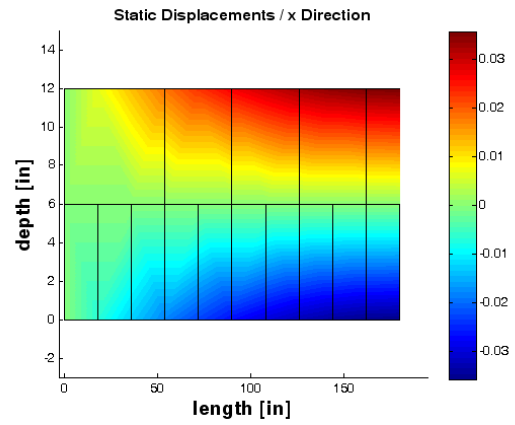


(b): Q4 Element Mesh 2

Figure 4-11: Q4 Beam Deflection Profile

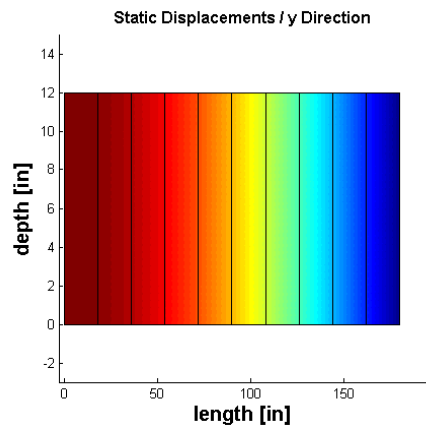


(a): Q4 Element Mesh 1

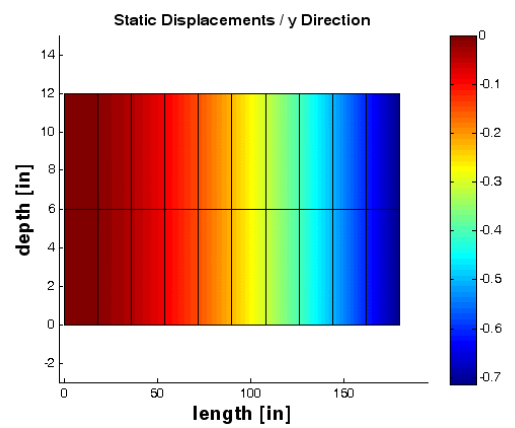


(b): Q4 Element Mesh 2

Figure 4-12: Contour Plot for Q4 Element Displacements in the x-Direction



(a): Q4 Element Mesh 1



(b): Q4 Element Mesh 2

Figure 4-13: Contour Plot for Q4 Element Displacements in the y-Direction

As seen in the CST elements, the deflection of the beam having Q4 elements behave in the same way due to the linear interpolation assumed in the element stiffness matrix. Table 4-4 illustrates the maximum tip deflection of the beam as -0.6895 inches for the first finite element mesh of 10 elements provided in sub-figures (a). Table 4-4 also provides the results from the alternative finite element mesh of 20 elements provided in sub-figures (b).

Finite Element Mesh	Tip Displacement (in)
10 Elements <i>Q4 Element Mesh 1</i>	-0.6895
10 Elements <i>Q4 Element Mesh 2</i>	-0.7140

Table 4-4: Maximum Tip Deflection for Q4 Elements

Like the CST, the Q4 element cannot exhibit pure bending. As for the CST elements, the linear interpolation used to quantify the shape functions exhibit element constraint. Refinement of the Q4 finite element mesh will help in convergence to the exact value but is not considered in this study.

4.5 Q8 Elements

The beam of Figure 4-1 is modeled using 10 Q8 elements. Figure 4-14 provides the finite element mesh generated by the Matlab program written as a result of this study. Each node is also labeled in Figure 4-14. DOF 1 – 6, located at nodes 1, 2 and 3, at the fixed end of the beam are constrained to zero displacement. The loading is applied to DOF 106, located at node 53, at the tip of the free end of the beam. The results of the analysis are shown in Figures 4-15 through 4-17. Figure 4-15 shows the displacement of the Q8 elements when a concentrated load is applied at the free end of a fixed beam. The

contour plots for the deflections are shown in Figures 4-15 and 4-17 for the x and y directions, respectively.

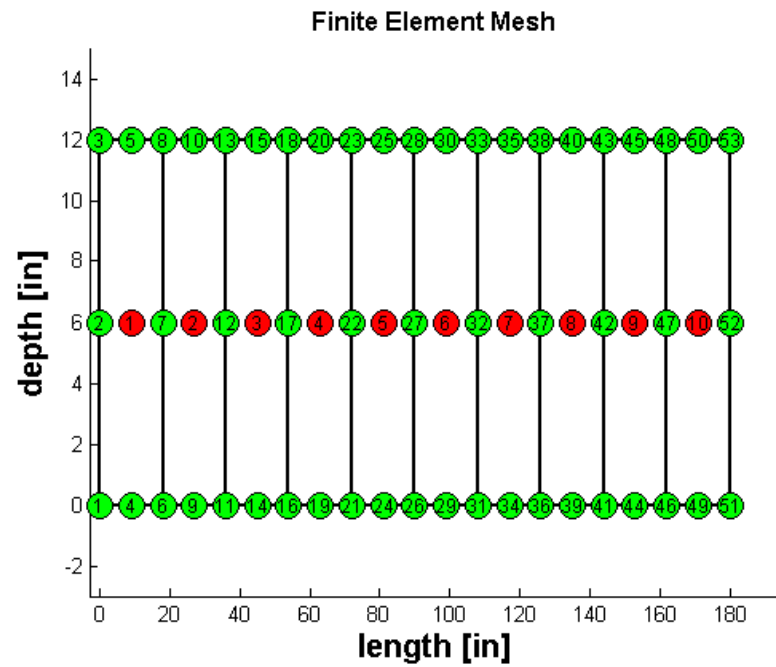


Figure 4-14: Q8 Element Mesh

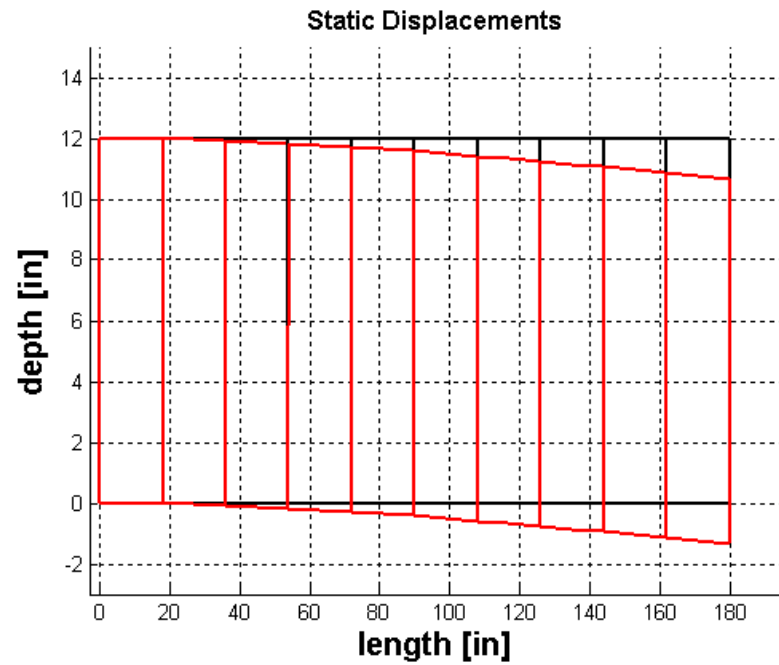


Figure 4-15: Q8 Beam Deflection Profile

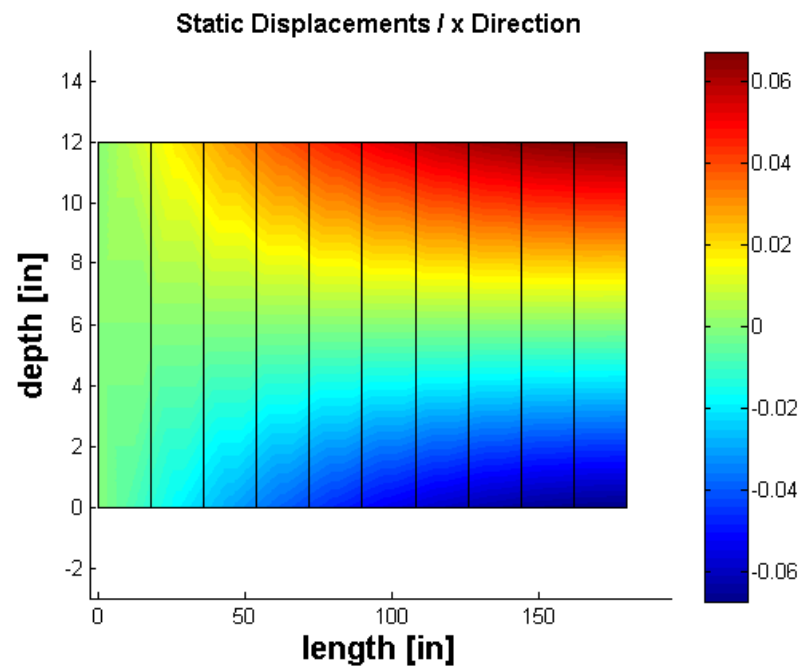


Figure 4-16: Contour Plot for Q8 Element Displacements in the x-Direction

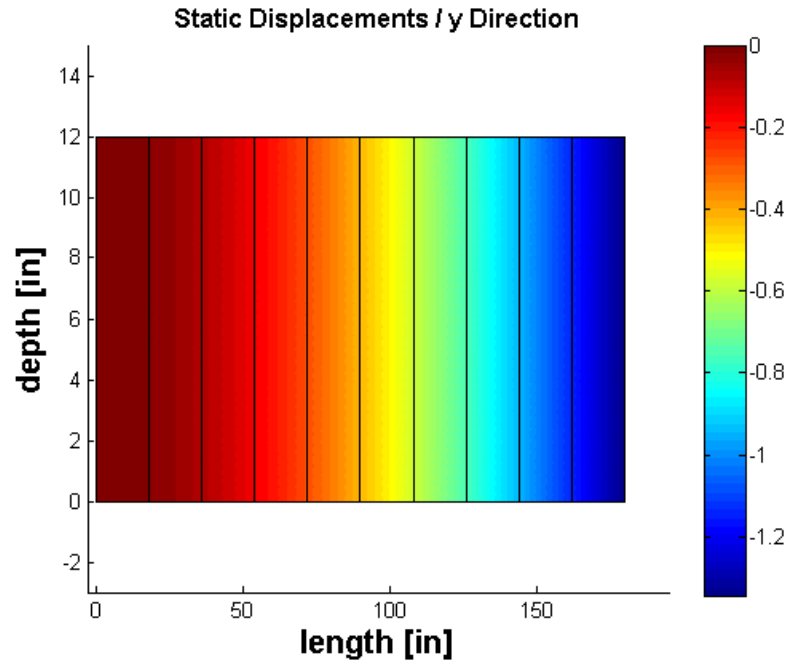


Figure 4-17: Contour Plot for Q8 Element Displacements in the y-Direction

As observed for the LST elements, the deflection of the beam is smooth when compared to the CST and Q4 elements. This is due to the quadratic interpolation of the Q8 elements. Figure 4-16 illustrates the deflection of the beam in the x direction. This deflection is from positive to negative x directions from top to the bottom nodes respectively. Table 4-5 illustrates the maximum tip deflection under the assigned load to be -1.3436in. As for the LST elements, the Q8 elements illustrate that fewer elements can be used when quadratic interpolation functions are used to formulate the element stiffness.

Finite Element Mesh	Tip Displacement (in)
10 Elements <i>Q8 Element Mesh</i>	-1.3436

Table 4-5: Maximum Tip Deflection for Q8 Elements

4.6 Comparison

Table 4-6 provides a quick comparison between the case studies presented in Sections 4.2 through 4.5. It is clear that the linearly interpolated elements are stiffer than the quadratically interpolated elements. Q4 elements show more convergence towards the analytical solution than the CST elements due to the higher number of DOF per element. While comparing the quadratic elements, it is noted that the Q8 element is more refined than the LST element. This is due to the fact that in the LST elements, γ_{xy} is represented as linear rather than quadratic in y . Hence, as seen in Table 4-6, LST and Q8 elements converge towards the analytical solution when compared to the CST and Q4 elements.

The analytical solution is determined from the maximum tip deflection, δ_{tip} , of the fixed beam formula given in equation 4-1. Here P is the load applied at the free end of the beam. L is the length of the beam and E is modulus of elasticity. I is the moment of inertia calculated from the rectangular cross section of the beam.

$$\delta_{tip} = \frac{PL^3}{3EI} \quad 4-1$$

Element Type	Tip Deflection (in)
CST <i>20 / 40 Elements</i>	-0.2328 / -0.3262
LST <i>20 Elements</i>	-1.3433
Q4 <i>10 / 20 Elements</i>	-0.6895 / -0.7140
Q8 <i>10 Elements</i>	-1.3436
Analytical	-1.3500

Table 4-6: Maximum Tip Deflection Comparison

CHAPTER V

CONCLUSION

5.1 Summary and Conclusions

The theoretical development for incorporating advanced finite elements is provided in this study. Both linear and quadratic interpolated elements for 2-D analysis are considered. Element stiffness matrices are developed using the shape functions which are derived from the displacement fields for a particular node in an element. A FEA code has been written including all the advanced elements using the concept of isoparametric elements. Gaussian quadrature is used as a tool for numerical integration and the final displacements are calculated using the Gauss elimination technique to solve the structural equations.

Deflections and deflected profiles are provided for a beam when a concentrated load is applied at the free end. The FEA program is capable of automatically generating an element mesh for different types of elements specified corresponding to the dimensions of the beam with applied boundary conditions. The deflected profile along with the contour plot for the beam in the x and y directions are provided. The maximum tip deflection of the beam when the same numbers of elements are used show that the CST

elements are much stiffer than the other elements. Q4 elements give relatively better results than the CST elements but are also stiff. LST and Q8 elements provide more accurate results than the other two types. This is due to the fact that the CST and Q4 elements use linear interpolation for their field displacements, whereas the LST and Q8 elements use quadratic interpolation. Refinement of the mesh increases the accuracy of deflection of the beam for all element types but was not considered in this study.

5.2 Recommendations

In addition to the work that was outlined in this study, the FEA code is also capable of analyzing thermal conductivity heat transfer through a material. The work on this heat transfer formulation is not complete at this time but it is recommended that the formulation be completed. When completed, temperature variations throughout the structural system will be able to be analyzed as a function of time to determine the degradation of structural properties due to such occurrences such as fire. This code can be improved to incorporate the heat conduction through the beam elements using dynamic heat analysis. Boundary conditions for heat transfer are applied to the beam so that the change in material properties of the beam can be determined with respect to time.

It is also recommended that the FEA program be improved to allow for more complex structures. Currently, the node input process is tedious and can be improved for structural systems. In doing so, coordinate transformations will be required to transform the local coordinates of the elements to the global coordinates of the system. The finite

element mesh can also be automatically refined so that the accuracy of deflection increases. Future studies can be conducted to incorporate a three-dimensional (3-D) brick elements as well as different boundary and loading conditions. Finally, it is also recommended that the calculation of stresses and strains be improved such that they are determined at the nodal locations of the finite element mesh.

Work done in conjunction with this study includes the incorporation of structural dynamic response due to dynamic loads (Saglar, 2009).

REFERENCES

- [1] Cook, R.D., Malkus, D.S., Plesha, M.E. and Witt, R.J. (2002), “*Concepts and Applications of Finite Element Analysis*”, John Wiley & Sons.
- [2] MacNeal, R.H. (1993), “*Finite Elements: their design and performance*”, Marcel Dekker, Inc.
- [3] Kwon, Y.W., Bang, H. (2000), “*The Finite Element Method using Matlab, Second Edition*”, CRC press.
- [4] Park, H.C. and Lee, S.W. (2001), “Cure Simulation of Thick Composite Structures Using the Finite Element Method, *Journal of Composite Materials*, Vol. 35, No. 03, pp. 188 - 201.
- [5] Ozcan, D.M., Bayraktar, A., Sahin, A., Haktanir, T. and Turker, T. (2008), “Experimental and Finite Element Analysis on the Steel Fiber-Reinforced Concrete (SFEC) Beams Ultimate Behavior”, *Journal of Construction and Building Materials*, Vol. 23, No. 02, pp. 1064 - 1077.
- [6] Barbato, M., Zona, A. and Conte, J.P. (2006), “Finite Element Response Sensitivity Analysis Using Three-Field Mixed Formulation: General Theory and Application to Frame Elements”, *Journal of International Journal for Numerical Methods in Engineering*, Vol. 69, No. 01, pp.114 - 161.
- [7] Salem, A.Z.I., Canaan, S.A. and Saigal, S. (2001), “Mid-Node Admissible Spaces for Quadratic Triangular 2D Finite Elements with One Edge Curved”, *Journal of*

- International Journal for Numerical Methods in Engineering*, Vol. 50, No. 01, pp. 181 - 197.
- [8] Paulino, G.H. and Lee, C.H. (2007), “A Modified Q4/Q4 Element for Topology Optimization”, *Journal of Structural and Multidisciplinary Optimization*, Vol. 37, No. 03, pp. 255 - 264.
 - [9] Li, Q.S., Yang, L.F. and Li, G.Q. (2001), “The Quadratic Finite Element/Strip with Generalized Degrees of Freedom and Their Application”, *Journal of Finite Element Analysis and Design*, Vol. 37, No. 04, pp. 325 - 339.
 - [10] Kikuchi, F., Okabe, M. and Fujio, H. (1999), “Modification of the 8-Node Serendipity Element”, *Journal of Finite Element Analysis Design*, Vol. 37, No. 04, pp. 325 - 339.
 - [11] Long, C.S. and Groenwold, A.A. (2004), “Reduced Modified Quadratures for Quadratic Membrane Finite Elements”, *International Journal for Numerical Methods in Engineering*, Vol. 61, No. 06, pp. 837 - 855.
 - [12] Alberty, J., Carstensen, C., Funken, S.A. and Klose, R. (2002), “Matlab Implementation of the Finite Element Method in Elasticity”, *Journal of Computing*, Vol. 69, No. 3, pp. 239 - 263.
 - [13] Bartels, S., Carstensen, C. and Hecht, A. (2004), “P2Q2Iso2D = 2D Isoparametric FEM in Matlab”, *Journal of Computation and Applied Mathematics*, Vol. 192, No. 2, pp. 219 - 250.
 - [14] Saglar, M.A. (2009), “*Development of a Finite Element Analysis Program for Structural Dynamics Applications*”, MS Thesis, Oklahoma State University.

APPENDIX A

MATLAB FEA PROGRAM

The Matlab FEA program that was developed during the course of this study is provided herein. First, the main FEA program (not containing the sub-routines since the FEA program is still under development) is provided. Second, the general input file used to setup an analysis is provided. Both of these files are Matlab script files and can be executed in all versions of Matlab. To generate graphical output, the full version of Matlab is required.

Main Program

```
% *****
% Finite Element Analysis (FEA) Program to Determine the Structural and Thermal Response of Structural Systems
% Written By: Muhammet Saglar & Rameez Iqbal (Advised By: Dr. Jonathan S. Goode)
% School of Civil and Environmental Engineering
% Oklahoma State University
% *****

% *****
% *****
% Begin Program

% Clear Variables and Settings
clear all

% Clear Screen
clc;

% Inhibit Warning Messages
warning('off','all');

disp('*****');
disp('Finite Element Analysis');
disp('Written by: Saglar/Iqbal (Goode)');
disp('School of Civil and Environmental Engineering');
disp('Oklahoma State University');
disp('*****');
```

```

disp('*****');
disp('Running Analysis .....');
disp('*****');

% Begin Timer
tic;

% *****
% Pre-Processing Section
% *****

disp('Pre-Processing .....');

% Call M-File for User Input Data from fea_input.m
disp('..... Obtaining Input .....');
fea_input;

% Call M-File for Initialization of Variables from fea_initialize.m
disp('..... Initializing Program .....');
fea_initialize;

% *****
% End Pre-Processing Section
% *****

% *****
% Processing Section
% *****

disp('Processing .....');

% -----
% Assembly Process

disp('..... Assembly Process .....');

% Determine Element Property Matrices (Constitutive, Mass, Conductivity)
[property_matrix_k,property_matrix_m,property_matrix_t] =
fea_property_matrix(analysis_type_material,material_E,material_v,material_rho,material_t);

% Determine Gauss Quadrature Sampling Points and Weights
[gauss_points,gauss_weights] = fea_gauss_2d(element_type);

% Initialize Global Index of DOF
index_global = zeros(edof,nel);

% Assemble System Matrices According to Analysis
if analysis_type == 'S'

    % Static Analysis

    disp('..... Static Analysis .....');
    disp('..... Building Stiffness .....');

    % Loop Through All Elements to Determine System Matrices (Stiffness, Mass, and Conductivity)
    % NOTE: Depending on the Analysis Type, Some Matrices Will Not be Generated
    for j = 1:nel

        % Call Sub-Routine to Extract Nodes for the j-th Element
        [element_nodes,element_xcoord,element_ycoord] = fea_node_coordinates(j,nel,node_connectivity,node_coord);

        % Call Sub-Routine to Extract System DOFs for the j-th Element
        index = fea_index_dof(element_nodes,nel,ndof,edof);

        % Save Index of DOF to Global Variable
        index_global(:,j) = index;

        % Call Sub-Routine to Generate Element Stiffness Matrix

```



```

        element_k =
fea_element_matrix_stiffness(element_type,gauss_points,gauss_weights,element_xcoord,element_ycoord,nnel,edof,property_matrix_
k);

        % Call Sub-Routine to Assemble System Stiffness Matrix
        system_k = fea_element_matrix_assemble(system_k,element_k,index,edof);

    end

elseif analysis_type == 'D'

    % Dynamic Analysis

    disp('..... Dynamic Analysis .....');
    disp('..... Building Stiffness and Mass .....');

    % Loop Through All Elements to Determine System Matrices (Stiffness, Mass, and Conductivity)
    % NOTE: Depending on the Analysis Type, Some Matrices Will Not be Generated
    for j = 1:nnel

        % Call Sub-Routine to Extract Nodes for the j-th Element
        [element_nodes,element_xcoord,element_ycoord] = fea_node_coordinates(j,nnel,node_connectivity,node_coord);

        % Call Sub-Routine to Extract System DOFs for the j-th Element
        index = fea_index_dof(element_nodes,nnel,ndof,edof);

        % Save Index of DOF to Global Variable
        index_global(:,j) = index;

        % Call Sub-Routine to Generate Element Stiffness Matrix
        element_k =
fea_element_matrix_stiffness(element_type,gauss_points,gauss_weights,element_xcoord,element_ycoord,nnel,edof,property_matrix_
k);

        % Call Sub-Routine to Assemble System Stiffness Matrix
        system_k = fea_element_matrix_assemble(system_k,element_k,index,edof);

        % Call Sub-Routine to Generate Element Mass Matrix
        element_m =
fea_element_matrix_mass(element_type,gauss_points,gauss_weights,element_xcoord,element_ycoord,nnel,edof,property_matrix_m);

        % Call Sub-Routine to Assemble System Mass Matrix
        system_m = fea_element_matrix_assemble(system_m,element_m,index,edof);

    end

end

% End Assembly Process
% -----

% -----
% Boundary and Loading Conditions

disp('..... Applying Boundary and Loading Conditions .....');

% Apply Boundary and Loading Conditions According to Analysis
if analysis_type == 'S'

    % Static Analysis

    % Call Sub-Routine to Apply Boundary and Loading Conditions
    [system_k,system_f] = fea_system_boundary_load(system_k,system_f,boundary_dof,boundary_dof_val,load_dof,load_dof_val);

elseif analysis_type == 'D'

    % Dynamic Analysis

    % Boundary and Loading Conditions Must Be Applied at Each Point in Time During the Dynamic Analysis
    % See the Analysis Section Below

```

```

end

% End Boundary Conditions
% -----

% -----
% Analysis

disp('..... Solving .....');

% Based on Analysis Type
if analysis_type == 'S'

    % Static Analysis

    disp('..... Static Analysis .....');

    % Solve System Equations for Static Displacements
    system_d = system_k \ system_f;

elseif analysis_type == 'D'

    % Dynamic Analysis

    disp('..... Dynamic Analysis .....');

    % Call Sub-Routine to Solve System Equations for Dynamic Displacements
    [system_d,system_d_vel,system_d_acc,system_f,dynamic_time] =
fea_dynamic_analysis(system_k,system_m,boundary_dof,boundary_dof_val,load_dof,load_file,dynamic_delta_t,dynamic_beta,dyna
mic_gamma,sdof);

end

% End Analysis
% -----

% *****
% End Processing Section
% *****

% *****
% Post-Processing Section
% *****

disp('Post-Processing .....');

% -----
% Plot Generation

% Plot Finite Element Mesh
if plot_element_mesh == 'Y'

    disp('..... Plotting Element Mesh .....');

    % Call Sub-Routine to Plot Finite Element Mesh
    plot_handle =
fea_plot_mesh(proj_name,nel,node_coord,node_connectivity,plot_element_mesh_numbers,plot_element_mesh_node_numbers);

end

% Based on Analysis Type
if analysis_type == 'S'

    % Static Analysis

    disp('..... Static Analysis .....');

    % Plot Static Displacements
    if plot_static_displacement == 'Y'

```

```

disp('..... Plotting Static Displacements .....');

% Call Sub-Routine to Plot Static Displacements
plot_handle = fea_plot_static_displacement(proj_name,nel,node_coord,node_connectivity,system_d,index_global);

if plot_static_contour_displacement == 'Y'

    % Call Sub-Routine to Plot Displacement Contours
    plot_handle = fea_plot_static_displacement_contour(proj_name,node_coord,node_connectivity,system_d,sdof,nel);

end

end

% Determine Static Stresses and Strains for Plots
if plot_static_stress == 'Y' || plot_static_strain == 'Y'

    disp('..... Determining Static Stresses and Strains .....');

    % Call Sub-Routine to Determine Stresses and Strains from Displacements
    [system_stress,system_strain,gauss_points_coord] =
fea_stress_strain(nel,nnel,edof,node_connectivity,node_coord,index_global,system_d,property_matrix_k,element_type,gauss_points);

end

% Plot Static Stresses
if plot_static_stress == 'Y'

    disp('..... Plotting Static Stresses .....');

    % Call Sub-Routine to Plot Static Stresses
    plot_handle = fea_plot_static_stress(proj_name,nel,node_coord,node_connectivity,system_stress,gauss_points_coord);

    if plot_static_contour_stress

        % Call Sub-Routine to Plot Stress Contours
        [plot_handle] =
fea_plot_static_stress_contour(proj_name,nel,node_coord,node_connectivity,system_stress,gauss_points_coord);

    end

end

% Plot Static Strains
if plot_static_strain == 'Y'

    disp('..... Plotting Static Strains .....');

    % Call Sub-Routine to Plot Static Strains
    plot_handle = fea_plot_static_strain(proj_name,nel,node_coord,node_connectivity,system_strain,gauss_points_coord);

    if plot_static_contour_strain

        % Call Sub-Routine to Plot Strain Contours
        [plot_handle] =
fea_plot_static_strain_contour(proj_name,nel,node_coord,node_connectivity,system_strain,gauss_points_coord);

    end

end

elseif analysis_type == 'D'

    % Dynamic Analysis

    disp('..... Dynamic Analysis .....');

    % Call Sub-Routine to Plot Dynamic Displacements (Movie)

```

```

        plot_handle =
fea_plot_dynamic_displacement(proj_name,nel,node_coord,node_connectivity,system_d,dynamic_time,index_global);

        disp('..... Plotting Dynamic Displacements .....');

    end

% *****
% End Post-Processing Section
% *****

disp('*****');
disp('Analysis Done .....');
disp('*****');

% End Timer
analysis_time = toc;

disp(['Total Time of Analysis: ',num2str(analysis_time),' seconds']);
disp('*****');

% End Program
% *****
% *****

```

Input File

```

% *****
% Finite Element Analysis (FEA) - Input File
% Written By: Muhammet Saglar & Rameez Iqbal (Advised By: Dr. Jonathan S. Goode)
% School of Civil and Environmental Engineering
% Oklahoma State University
% *****

% *****
% *****
% Begin Input File

% -----
% Project Name

% Define Project Name
% NOTE: Automatically Saved Figures and Data Will Be Proceeded with the Project Name
proj_name = 'test';

% -----
% Analysis Options

% Define Analysis Type
% Analysis Type Options:
% Input 'S' = Static Analysis
% Input 'D' = Dynamic Analysis
% Input 'T' = Thermal Analysis
analysis_type = 'S';

% -----
% Input Finite Element Mesh Properties

% Input Element Type
% Input 'Q4' for Bilinear Rectangular Element
% Input 'Q8' for Quadratic Rectangular Element
% Input 'CST' for Constant Strain / Linear Triangular Element
% Input 'LST' for Linear Strain / Quadratic Triangular Element

```

```

element_type = 'Q4';

% Input Node Coordinates (x and y coordinates) [Note Units => Length]
node_coord = [ 0.0  0.0  ;...
               12.0 0.0  ;...
               24.0 0.0  ;...
               36.0 0.0  ;...
               48.0 0.0  ;...
               60.0 0.0  ;...
               72.0 0.0  ;...
               84.0 0.0  ;...
               96.0 0.0  ;...
               108.0 0.0 ;...
               120.0 0.0 ;...
               0.0 12.0 ;...
               12.0 12.0 ;...
               24.0 12.0 ;...
               36.0 12.0 ;...
               48.0 12.0 ;...
               60.0 12.0 ;...
               72.0 12.0 ;...
               84.0 12.0 ;...
               96.0 12.0 ;...
               108.0 12.0 ;...
               120.0 12.0 ];

% Input Nodal Connectivity for Each Element (CCW from Bottom-Left)
node_connectivity = [ 1  2  13 12  ;...
                     2  3  14 13  ;...
                     3  4  15 14  ;...
                     4  5  16 15  ;...
                     5  6  17 16  ;...
                     6  7  18 17  ;...
                     7  8  19 18  ;...
                     8  9  20 19  ;...
                     9  10 21 20  ;...
                     10 11 22 21  ];

% -----
% Input Boundary Conditions

% Input DOF Constrained
boundary_dof = [1 2 23 24];

% Input Constrained DOF Prescribed Values [Note Units => Length]
% NOTE: For Dynamic Analysis, All Prescribed Values MUST BE Zero (0)
boundary_dof_val = [0 0 0 0];

% -----
% Input Material Properties

% Material Properties
% NOTE: If the property is not being used, enter 0 (zero)

% Modulus of Elasticity [Note Units => Force / Length^2]
material_E = 1e6;

% Poisson's Ratio [Unitless]
material_v = 0.3;

% Mass Density [Note Units => Mass / Length^3]
material_rho = 1;

% Define Material Analysis Type
% Material Analysis Type Options:
% Input '1' = Plane Stress Analysis (2-D)
% Input '2' = Plane Strain Analysis (2-D)
% Input '3' = Three-Dimensional Analysis (3-D)
analysis_type_material = 1;

```

```

% -----
% Input Load Properties

% Static Loading (If Applicable)
% NOTE: If Static Analysis is not being considered, NO changes are needed

if strcmp(analysis_type,'S') == 1

    % Input DOF Statically Loaded
    load_dof = [18 20];

    % Input DOF Statically Loaded Prescribed Values [Note Units => Force]
    load_dof_val = [-500 -500];

% Dynamic Loading (If Applicable)
% NOTE: If Dynamic Analysis is not being considered, NO changes are needed

elseif strcmp(analysis_type,'D') == 1
    % Input DOF Dynamically Loaded
    load_dof = [44];

    % Input Load Time-Series File Name [Note Units => Force]
    % NOTE: First Number Designates the Load at Time = 0 seconds
    % Column 1 -> n = DOFs
    % Row 1 -> n = Load at Time Increments
    load_file = 'load.dat';

    % Input Time Step of Load Time-Series [seconds]
    dynamic_delta_t = 0.1;

    % Define Newmark-Beta Parameters [Unitless]
    % Average Acceleration Method => Beta = 0.25 & Gamma = 0.5 (Unconditionally Stable)
    % Linear Acceleration Method => Beta = 0.16667 & Gamma = 0.5 (Conditionally Stable -> delta_t / Tn <= 0.551)
    dynamic_beta = 0.25;
    dynamic_gamma = 0.5;

end

% -----
% Output Options

% General Plots

% Plot Element Mesh (Y or N)
plot_element_mesh = 'Y';

% Plot Element Mesh Options

% Display Element Numbers on Element Mesh Plot (Y or N)
plot_element_mesh_numbers = 'Y';

% Display Element Node Numbers on Element Mesh Plot (Y or N)
plot_element_mesh_node_numbers = 'Y';

% Static Analysis (If Applicable)
% NOTE: If Static Analysis is not being considered, NO changes are needed

% Plot Static Displacements (Y or N)
plot_static_displacement = 'Y';

% Contour Plots of Displacements (Y or N)
% NOTE: Must Plot Static Displacements to Plot Contours
plot_static_contour_displacement = 'Y';

% Plot Static Stresses (Y or N)
plot_static_stress = 'N';

% Contour Plots of Stresses (Y or N)
% NOTE: Must Plot Static Stresses to Plot Contours
plot_static_contour_stress = 'Y';

```

```

% Plot Static Strains (Y or N)
plot_static_strain = 'N';

% Contour Plots of Strains (Y or N)
% NOTE: Must Plot Static Strains to Plot Contours
plot_static_contour_strain = 'Y';

% Dynamic Analysis (If Applicable)
% NOTE: If Dynamic Analysis is not being considered, NO changes are needed

% Movie of Dynamic Displacements (Y or N)
movie_dynamic_displacement = 'Y';

% End Input File
% *****
% *****

```

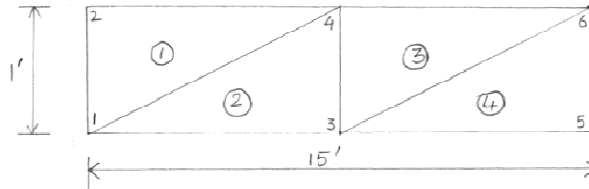
APPENDIX B

CONFIRMATION CALCULATIONS OF ELEMENT FORMULATIONS

Appendix B provides confirmation calculations of the element stiffness matrix for each of the element considered in this study. These calculations were done by hand and then compared to the resulting stiffness matrices and displacements developed by the Matlab FEA program as provided in Appendix A. These calculations were completed to ensure the Matlab FEA program was making calculations correctly, not necessarily for accuracy with respect to any given problem.

Constant Strain Triangular (CST) Element

CONSTANT STRAIN TRIANGULAR ELEMENT



$$t = 1$$

$$\nu = 0.3$$

$$E = 10^6 \text{ lb/in}^2$$

CST Element 1

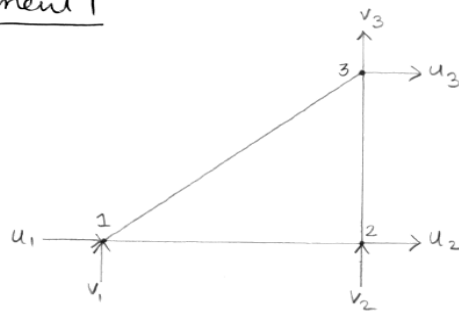


Figure - 1

Field quantity for CST element is given by

$$\phi = a_1 + a_2x + a_3y$$

$$u_1 = a_1$$

$$u_2 = a_1 + a_2 x_2$$

$$u_3 = a_1 + a_2 x_3 + a_3 y_3$$

Solving the above equations give

$$a_1 = u_1$$

$$a_2 = \frac{u_2 - u_1}{x_2}$$

$$a_3 = \frac{(x_3 - x_2)}{x_2 y_3} u_1 - \frac{x_3}{x_2 y_3} u_2 + \frac{1}{y_3} u_3$$

From the above interpolated field quantities,
the strain-displacement matrix can be
computed as

$$[B] = \begin{bmatrix} -\frac{1}{x_2} & 0 & \frac{1}{x_2} & 0 & 0 & 0 \\ 0 & \frac{x_3 - x_2}{x_2 y_3} & 0 & \frac{-x_3}{x_2 y_3} & 0 & \frac{1}{y_3} \\ \frac{x_3 - x_2}{x_2 y_3} & -\frac{1}{x_2} & \frac{-x_3}{x_2 y_3} & \frac{1}{x_2} & \frac{1}{y_3} & 0 \end{bmatrix}$$

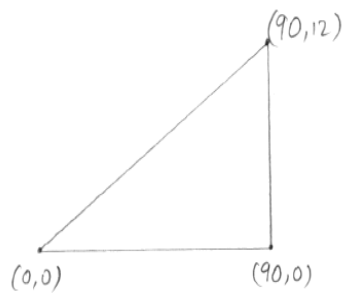


Figure 2

From figure 2, the coordinates for the nodes can be computed as

$$(x_1, y_1) = (0, 0)$$

$$(x_2, y_2) = (90, 0)$$

$$(x_3, y_3) = (90, 12)$$

Substituting the nodal coordinates in the strain-displacement matrix we get

$$[B] = \begin{bmatrix} -0.011 & 0 & 0.011 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.083 & 0 & 0.0833 \\ 0 & -0.011 & -0.083 & 0.011 & 0.0833 & 0 \end{bmatrix}$$

From the given modulus of elasticity and poisson's ratio, the constitutive matrix can be computed as

$$[E] = \frac{10^6}{0.91} \begin{bmatrix} 1 & 0.3 & 0 \\ 0.3 & 1 & 0 \\ 0 & 0 & 0.35 \end{bmatrix}$$

The element stiffness matrix can be computed from

$$[k] = [B]^T * [E] * [B] * t * A$$

			<u>Local Stiffness Matrix</u>			
	1442308	5.58E-11	-1.14E-11	-192307.6923	-1442307.692	192307.6923
	5.58E-11	4120879.121	-164835.1648	-4.00E-12	164835.1648	-4120879.121
k =	-1.14E-11	-164835.1648	73260.07326	0	-73260.07326	164835.1648
	-192308	-4.00E-12	0	25641.02564	192307.6923	-25641.02564
	-1442308	164835.1648	-73260.07326	192307.6923	1515567.766	-357142.8571
	192307.7	-4120879.121	164835.1648	-25641.02564	-357142.8571	4146520.147

CST element 2

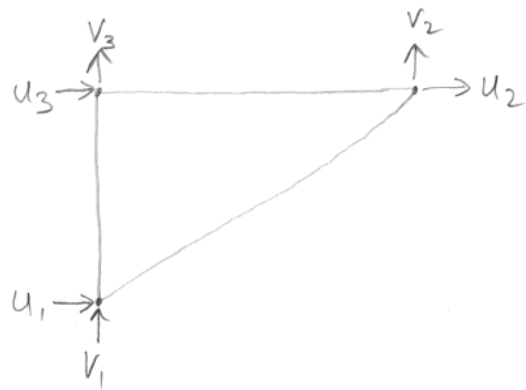


Fig-3

Field quantity for the element from Fig-3 is calculated as

$$a_1 = u_1$$

$$a_2 = \frac{y_2 - y_3}{x_2 y_3} u_1 + \frac{1}{x_2} u_2 - \frac{y_2}{x_2 y_3} u_3$$

$$a_3 = -\frac{1}{y_3} u_1 - \frac{1}{y_3} u_3$$

From the above interpolated field quantities, the strain-displacement matrix can be computed as

$$[B] = \begin{bmatrix} \frac{y_2 - y_3}{x_2 y_3} & 0 & \frac{1}{x_2} & 0 & \frac{-y_2}{x_2 y_3} & 0 \\ 0 & -\frac{1}{y_3} & 0 & 0 & 0 & \frac{1}{y_3} \\ -\frac{1}{y_3} & \frac{y_2 - y_3}{x_2 y_3} & 0 & \frac{1}{x_2} & \frac{1}{y_3} & \frac{-y_2}{x_2 y_3} \end{bmatrix}$$

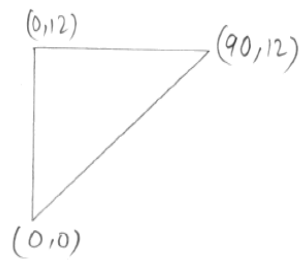


Figure - 4

From figure 4, the coordinates of the nodes can be computed as

$$(x_1, y_1) = (0, 0)$$

$$(x_2, y_2) = (90, 12)$$

$$(x_3, y_3) = (0, 12)$$

Substituting the nodal coordinates in strain displacement matrix, we get

$$[B] = \begin{bmatrix} 0 & 0 & 0.011 & 0 & -0.011 & 0 \\ 0 & -0.083 & 0 & 0 & 0 & 0.083 \\ -0.083 & 0 & 0 & 0.011 & 0.083 & -0.011 \end{bmatrix}$$

The element stiffness matrix can be computed as

	<u>Local Stiffness Matrix</u>					
	73260.1	0	-73260	164835.2	0.0	-164835.2
	0	25641.0	192307.7	-25641	-192307.7	0.0
K =	-73260	192307.7	1515567.8	-357143	-1442307.7	164835.2
	164835.2	-25641	-357143	4146520.1	192307.7	-4120879.1
	0.0	-192307.7	-1442307.7	192307.7	1442307.7	0.0
	-164835.2	0.0	164835.2	-4120879.1	0.0	4120879.1

The local stiffness element matrix for CST element 1 and CST element 2 are mapped into the global coordinates

The displacement of the beam is determined by Gauss-Elimination method.

The displacements are found as

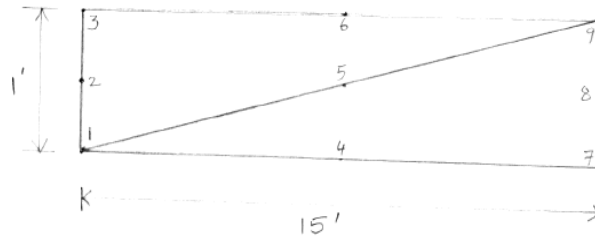
Global Stiffness Matrix

D.O.F	1	2	3	4	5	6	7	8	9	10	11	12
1	1.52E+36	5.58E-11	-1442308	192307.7	-73260.1	164835.2	-1.14E-11	-357142.9	0	0	0	0
2	5.58E-11	4.15E+36	164835.2	-4120879	192307.7	-25641	-357143	-4.00E-12	0	0	0	0
3	-1442308	164835.2	1.52E+36	-357143	0	0	-73260.1	192307.69	0	0	0	0
4	192307.69	-4120879	-357142.9	4.15E+36	0	0	164835.2	-25641.03	0	0	0	0
5	-73260.07	192307.7	0	0	3031136	-357143	-2884615	357142.86	-73260.1	164835	-1.14E-11	-357143
6	164835.16	-25641	0	0	-357143	8293040	357142.9	-8241758	192308	-25641	-357143	-4.00E-12
7	-1.14E-11	-357143	-73260.07	164835.2	-2884615	357142.9	3031136	-357142.9	0	0	-73260.1	192307.7
8	-357142.9	-4.00E-12	192307.7	-25641	357142.9	-8241758	-357143	8293040.3	0	0	164835.2	-25641
9	0	0	0	0	-73260.1	192307.7	0	0	1515568	-357143	-1442308	164835.2
10	0	0	0	0	164835.2	-25641	0	0	-357143	4146520	192307.7	-4120879
11	0	0	0	0	-1.14E-11	-357143	-73260.1	164835.16	-1442308	192308	1515568	0
12	0	0	0	0	-357143	-4.00E-12	192307.7	-25641.03	164835	-4E+06	0	4146520

<u>Beam Displacement Comparison</u>					
	Hand Calculations			Matlab FEA Program	
	D.O.F	Displacement		D.O.F	Displacement
	1	0		1	0
	2	0		2	0
	3	0		3	0
	4	0		4	0
	5	-0.000770679		5	-0.000770679
	6	-0.00754882		6	-0.00754882
	7	0.000719295		7	0.000719295
	8	-0.007525983		8	-0.007525983
	9	-0.001054455		9	-0.001054455
	10	-0.022512619		10	-0.022512619
	11	0.000927531		11	0.000927531
	12	-0.022501883		12	-0.022501883

Linear Strain Triangular (LST) Elements

LINEAR STRAIN TRIANGULAR ELEMENT



$$t = 1$$

$$\nu = 0.3$$

$$E = 10^6 \text{ lb/in}^2$$

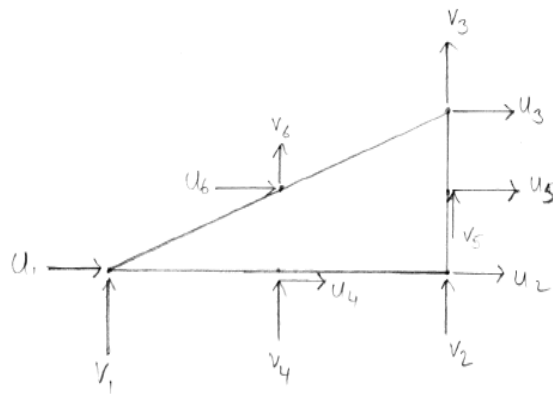


Figure 1

For the LST element shown in figure 1,
in terms of generalized DOF a_i , its displacement
field and associated strain fields are

$$u = a_1 + a_2 x + a_3 y + a_4 x^2 + a_5 xy + a_6 y^2$$

$$v = a_7 + a_8 x + a_9 y + a_{10} x^2 + a_{11} xy + a_{12} y^2$$

$$\epsilon_x = a_2 + 2a_4 x + a_5 y$$

$$\epsilon_y = a_9 + a_{11} x + 2a_{12} y$$

$$\gamma_{xy} = (a_3 + a_8) + (a_5 + 2a_{10})x + (2a_6 + a_{11})y$$

The shape functions are found from the
displacement field as

$$N_1 = 1 - 0.0167x + 0.0001x^2$$

$$N_2 = 0.0222x - 0.3333y - 0.0001x^2 + 0.0019xy$$

$$N_3 = -0.0056 X + 0.0833 Y + 0.0001 X^2 - 0.0019 XY \\ + 0.0139 Y^2$$

$$N_4 = 0.0019 XY - 0.0278 Y^2$$

$$N_5 = -0.0833 Y + 0.0139 Y^2$$

$$N_6 = 0.3333 Y - 0.0019 XY$$

The strain-displacement matrix can be found from $[B] = [D] [N]$. $[B]$ is a 3×12 matrix. The element stiffness matrix is given by

$$[K]_{12 \times 12} = \iint \underset{12 \times 3}{[B]^T} \underset{3 \times 3}{[E]} \underset{3 \times 12}{[B]} t \, dx \, dy$$

Figure 2 shows the coordinates of the LST element.

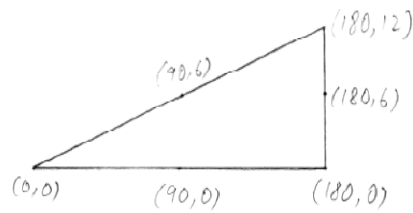


Figure 2

The limits for the integration in x is from 0 to 180. Whereas, the equation is integrated in y with respect to the equation of the hypotenuse line from 0.

The local element stiffness matrix is determined for the LST element as.

Local Stiffness Matrix

DOF	1	2	3	4	5	6	7	8	9	10	11	12
1	36630	0	-48840	219780	12210	-54945	0	0	0	54945	0	-219780
2	0	12821	256410	-17094	-64103	4274	0	0	64103	0	-256410	0
3	-48840	256410	7789988	-476190	-48840	219780	0	-476190	0	0	-7692308	476190
4	219780	-17094	-476190	22012210	256410	-17094	-476190	0	0	0	476190	-21978022
5	12210	-64103	-48840	256410	2921245	-357143	-3846154	219780	961538	-54945	0	0
6	-54945	4274	219780	-17094	-357143	8254579	256410	-10989011	-64103	2747253	0	0
7	0	0	0	-476190	-3846154	256410	7789988	-476190	-3846154	219780	-97680	476190
8	0	0	-476190	0	219780	-10989011	-476190	22012210	256410	-10989011	476190	-34188
9	0	64103	0	0	961538	-64103	-3846154	256410	2884615	0	0	-256410
10	54945	0	0	0	-54945	2747253	219780	-10989011	0	8241758	-219780	0
11	0	-256410	-7692308	476190	0	0	-97680	476190	0	-219780	7789988	-476190
12	-219780	0	476190	-21978022	0	0	476190	-34188	-256410	0	-476190	22012210

Similarly, the element stiffness matrix for the second element is determined. The element stiffness matrix for both the elements are mapped to the global element stiffness matrix

The displacements are found from the global element stiffness matrix as

Local Stiffness Matrix

DOF	1	2	3	4	5	6	7	8	9	10	11	12
1	192308	0	-256410	-256410	64103	64103	0	0	0	-64103	0	256410
2	0	549451	-219780	-732601	54945	183150	0	0	-54945	0	219780	0
3	-256410	-219780	1978022	476190	-256410	-256410	0	476190	0	0	-1465201	-476190
4	-256410	-732601	476190	1978022	-219780	-732601	476190	0	0	0	-476190	-512821
5	64103	54945	-256410	-219780	741758	357143	-732601	-256410	183150	64103	0	0
6	64103	183150	-256410	-732601	357143	741758	-219780	-256410	54945	64103	0	0
7	0	0	0	476190	-732601	-219780	1978022	476190	-732601	-256410	-512821	-476190
8	0	0	476190	0	-256410	-256410	476190	1978022	-219780	-256410	-476190	-1465201
9	0	-54945	0	0	183150	54945	-732601	-219780	549451	0	0	219780
10	-64103	0	0	0	64103	64103	-256410	-256410	0	192308	256410	0
11	0	219780	-1465201	-476190	0	0	-512821	-476190	0	256410	1978022	476190
12	256410	0	-476190	-512821	0	0	-476190	-1465201	219780	0	476190	1978022

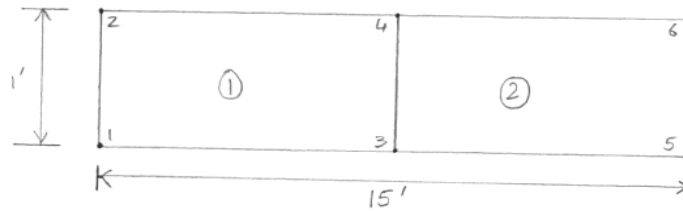
Global Stiffness Matrix

DOF	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	2.9E+36	-6E-11	-4E+06	256410	961538	-64103	-48840	219780	1E-09	-476190	-1E-09	7E-12	12210	-54945	-2E-10	1E-11	1E-10	1E+05
2	-6E-11	8E+36	219780	-1E+07	-54945	3E-06	256410	-17094	-476190	3.1E-09	1.5E-11	-3E-09	-64103	4273.5	1E-11	-5E-10	1E+05	4E-10
3	-4E+06	219780	7.8E+36	-476190	-4E+06	3E-05	0	0	-97680	476190	1.4E-09	-476190	0	0	0	0	4E-12	2E-11
4	256410	-1E+07	-476190	2.2E+37	219780	-1E+07	0	0	476190	-34188	-476190	3E-09	0	0	0	0	1E-11	7E-13
5	961538	-54945	-4E+06	219780	3E+36	-4E+05	0	0	2E-10	2.9E-11	-48840	256410	0	0	0	0	12210	-64103
6	-64103	3E+06	256410	-1E+07	-4E+05	8E+36	0	0	3E-11	9.3E-10	219780	-17094	0	0	0	0	-54945	4274
7	-48840	256410	0	0	0	0	8E+06	-476190	-8E+06	476190	0	0	-48840	219780	2E-10	-5E+05	-2E-10	3E-11
8	219780	-17094	0	0	0	0	-5E+05	2.2E+07	476190	-2E+07	0	0	256410	-17094	-5E+05	5E-10	3E-11	-9E-10
9	1.3E-09	-5E+05	-97680	476190	0	0	-8E+06	476190	2E+07	-952381	-8E+06	476190	-9E-11	-3E-11	-97680	476190	-5E-10	-5E+05
10	-476190	3E-09	476190	-34188	0	9E-10	476190	-2E+07	-952381	4.4E+07	476190	-2E+07	-3E-11	3E-10	476190	-34188	-5E+05	-1E-09
11	-1E-09	1E-11	1.2E-09	-476190	-48840	2E+05	0	0	-8E+06	476190	7789988	-476190	0	0	0	0	-48840	3E+05
12	7.3E-12	-3E-09	-476190	3.3E-09	256410	-17094	0	0	476190	-2E+07	-476190	2E+07	0	0	0	0	2E+05	-17094
13	12210	-64103	0	0	0	0	-48840	256410	-9E-11	-4E-11	0	0	3E+06	-357143	-4E+06	219780	1E+06	-54945
14	-54945	4273.5	0	0	0	0	219780	-17094	-3E-11	3.5E-10	0	0	-357143	8E+06	256410	-1E+07	-64103	3E+06
15	-2E-10	1E-11	0	0	0	0	2E-10	-476190	-97680	476190	0	0	-4E+06	256410	8E+06	-5E+05	-4E+06	2E+05
16	1.5E-11	-5E-10	0	0	0	0	-5E+05	4.7E-10	476190	-34188	0	0	219780	-1E+07	-5E+05	2E+07	3E+05	-1E+07
17	1.3E-10	119048	4.1E-12	1.5E-11	12210	-54945	-2E-10	2.6E-11	-5E-10	-476190	-48840	219780	961538	-64103	-4E+06	256410	3E+06	0
18	119048	4E-10	2.2E-11	6.8E-13	-64103	4274	3E-11	-9E-10	-476190	-1E-09	256410	-17094	-54945	3E+06	219780	-1E+07	0	8E+05

Beam Displacement Comparison					
		Hand-Calculations		Matlab FEA Program	
		DOF	Displacement	DOF	Displacement
		1	0	1	0
		2	0	2	0
		3	0	3	0
		4	0	4	0
		5	0	5	0
		6	0	6	0
		7	-0.03334747	7	-0.03334747
		8	-0.25289291	8	-0.25289291
		9	-2.91E-05	9	-2.91E-05
		10	-0.25255534	10	-0.25255534
		11	0.033269796	11	0.033269796
		12	-0.25287669	12	-0.25287669
		13	-0.06588415	13	-0.06588415
		14	-0.99964157	14	-0.99964157
		15	-1.00E-05	15	-1.00E-05
		16	-0.99933758	16	-0.99933758
		17	0.065865831	17	0.065865831
		18	-0.99969349	18	-0.99969349

Bilinear Rectangular (Q4) Element

BILINEAR RECTANGLE



$$t = 1$$

$$\nu = 0.3$$

$$E = 10^6 \text{ lb/in}^2$$

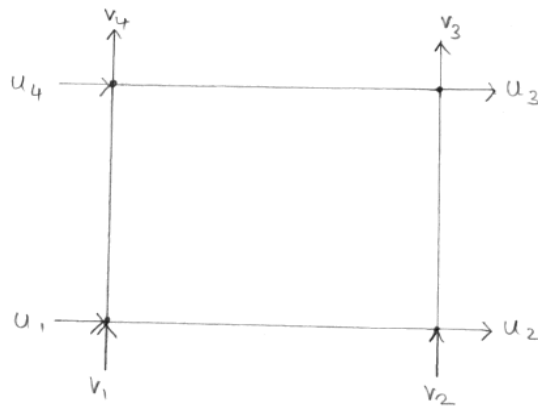


Figure 1

For the Q4 element shown in fig-1,
in terms of generalized d.o.f a_i , its
displacement field and associated strain field
are :

$$u = a_1 + a_2x + a_3y + a_4xy$$

$$v = a_5 + a_6x + a_7y + a_8xy$$

$$\epsilon_x = a_2 + a_4y$$

$$\epsilon_y = a_7 + a_8x$$

$$\gamma_{xy} = (a_3 + a_6) + a_4x + a_8y$$

By applying Lagrange's interpolation formula,
shape functions for the Q4 element can be
determined as

$$N_1 = \frac{(a-x)(b-y)}{4ab}$$

$$N_2 = \frac{(a+x)(b-y)}{4ab}$$

$$N_3 = \frac{(a+x)(b+y)}{4ab}$$

$$N_4 = \frac{(a-x)(b+y)}{4ab}$$

From $[B] = [d][N]$, strain displacement matrix is

$$[B] = \frac{1}{4ab} \begin{bmatrix} -(b-y) & 0 & (b-y) & 0 & (b+y) & 0 & -(b+y) & 0 \\ 0 & -(a-x) & 0 & -(a+x) & 0 & (a+x) & 0 & (a-x) \\ (-a-x) & -(b-y) & -(a+x) & (b-y) & (a+x) & (b+y) & (a-x) & -(b+y) \end{bmatrix}$$

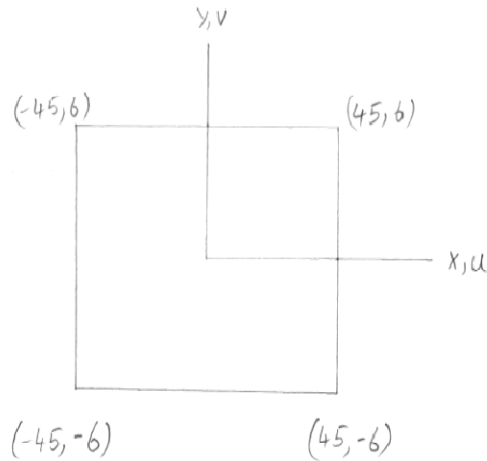


Figure 2

From figure 2, a and b can be found as

$$a = +45$$

$$b = 6$$

Substituting the values of a and b in $[B]$

$$[B] = \frac{1}{1080} \begin{bmatrix} -(6-y) & 0 & (6-y) & 0 & (6+y) & 0 & -(6+y) & 0 \\ 0 & -(45-x) & 0 & -(45+x) & 0 & (45+x) & 0 & (45-x) \\ -(45-x) & -(6-y) & -(45+x) & (6-y) & (45+x) & (6+y) & (45-x) & -(6+y) \end{bmatrix}$$

Substituting $[B]$ and $[E]$ in $[K]$

$$[K] = \int_{-b}^b \int_{-a}^a [B]^T [E] [B] t \, dx \, dy$$

The local stiffness matrix for the Q_4 element is found as

Local Stiffness Matrix

$$K = \begin{bmatrix} 1010378.51 & 178571.429 & 431929.2 & -13736.264 & -505189.26 & -178571.43 & -937118 & 13736.26 \\ 178571.429 & 2764346.76 & 13736.26 & 1356532.4 & -178571.43 & -1382173.4 & -13736.3 & -2738706 \\ 431929.182 & 13736.2637 & 1010379 & -178571.43 & -937118.44 & -13736.264 & -505189 & 178571.4 \\ -13736.2637 & 1356532.36 & -178571.4 & 2764346.8 & 13736.2637 & -2738705.7 & 178571.4 & -1382173 \\ -505189.255 & -178571.43 & -937118.4 & 13736.264 & 1010378.51 & 178571.429 & 431929.2 & -13736.3 \\ -178571.429 & -1382173.4 & -13736.26 & -2738705.7 & 178571.429 & 2764346.76 & 13736.26 & 1356532 \\ -937118.437 & -13736.264 & -505189.3 & 178571.43 & 431929.182 & 13736.2637 & 1010379 & -178571 \\ 13736.2637 & -2738705.7 & 178571.4 & -1382173.4 & -13736.264 & 1356532.36 & -178571 & 2764347 \end{bmatrix}$$

The local D.o.f for each element is mapped into the global d.o.f and the global stiffness matrix is found as

Global Stiffness Matrix

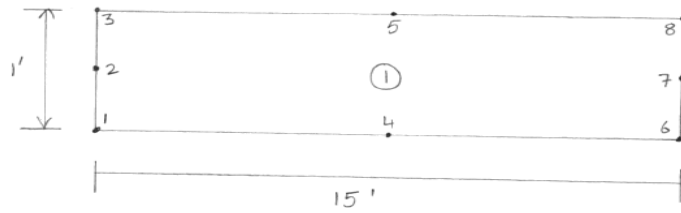
	1	2	3	4	5	6	7	8	9	10	11	12
1	1E+36	178571	-937118	13736	431929	-13736	-505189.2552	-178571.43	0	0	0	0
2	178571	2.76E+36	-13736	-2738706	13736	1356532	-178571.4286	-1382173.4	0	0	0	0
3	-937118	-13736	1E+36	-178571	-505189	178571	431929	13736	0	0	0	0
4	13736	-2738706	-178571	2.8E+36	178571	-1382173	-13736	1356532.36	0	0	0	0
5	431929	13736	-505189	178571	2020757.021	0	-1874236.874	0	431929	-13736	-505189.3	-178571
6	-13736	1356532	178571	-1382173	0	5528694	0	-5477411.5	13736	1356532	-178571.4	-1382173
7	-505189	-178571	431929	-13736	-1874236.874	0	2020757.021	0	-505189	178571	431929.18	13736
8	-178571	-1382173	13736	1356532	0	-5477411	0	5528693.53	178571	-1382173	-13736	1356532
9	0	0	0	0	431929	13736	-505189.2552	178571	1010379	-178571	-937118.4	-13736
10	0	0	0	0	-13736	1356532	178571	-1382173.4	-178571	2764347	13736	-2738706
11	0	0	0	0	-505189	-178571	431929	-13736	-937118	13736	1010378.5	178571
12	0	0	0	0	-178571	-1382173	13736	1356532.36	-13736	-2738706	178571	2764347

The displacements of the beam are found using Gauss-Elimination method. The final displacement of the beam is found as

Beam Dispalcement Comparison					
	Hand-Calculations			Matlab FEA Program	
	D.O.F	Displacement		D.O.F	Displacement
	1	0		1	0
	2	0		2	0
	3	0		3	0
	4	0		4	0
	5	-0.002232547		5	-0.002232547
	6	-0.018654177		6	-0.018654177
	7	0.002221229		7	0.002221229
	8	-0.018649147		8	-0.018649147
	9	-0.002955971		9	-0.002955971
	10	-0.059561303		10	-0.059561303
	11	0.002982398		11	0.002982398
	12	-0.059583108		12	-0.059583108

Quadratic Rectangular (Q8) Element

QUADRATIC RECTANGLE



$$t = 1$$

$$\nu = 0.3$$

$$E = 10^6 \text{ lb/in}^2$$

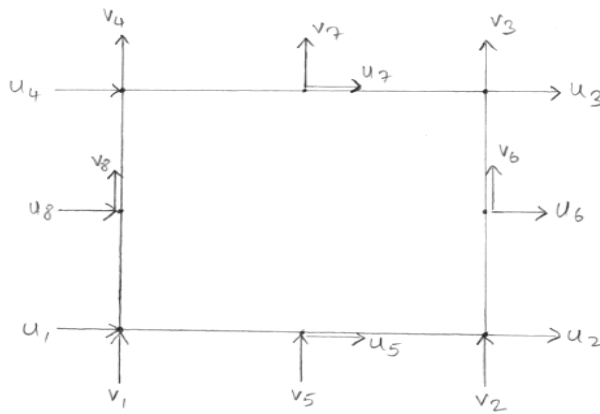


Figure 1

For the Q_8 element shown in figure 1, in terms of generalized d.o.f a_i , its displacement field and associated strain fields are:

$$u = a_1 + a_2 x + a_3 y + a_4 x^2 + a_5 xy + a_6 y^2 + a_7 x^2 y + a_8 xy^2$$

$$v = a_9 + a_{10} x + a_{11} y + a_{12} x^2 + a_{13} xy + a_{14} y^2 + a_{15} x^2 y + a_{16} xy^2$$

$$\epsilon_x = a_2 + 2a_4 x + a_5 y + 2a_7 xy + a_8 y^2$$

$$\epsilon_y = a_{11} + a_{13} x + 2a_{14} y + a_{15} x^2 + 2a_{16} xy$$

$$\gamma_{xy} = (a_3 + a_{10}) + (a_5 + 2a_{12})x + (2a_6 + a_{13})y + a_7 x^2 + 2(a_8 + a_{15})xy + a_{16} y^2.$$

The shape functions are found from the displacement field as

$$N_1 = 1 - 0.0167x - 0.25y + 0.0139y^2$$

$$N_2 = 0.3333y - 0.0019xy - 0.0278y^2$$

$$N_3 = -0.0833 Y + 0.0139 Y^2$$

$$N_4 = 0.0222 X$$

$$N_5 = 0.0019 XY$$

$$N_6 = -0.0056 X - 0.0011 XY$$

$$N_7 = 0.0019 XY$$

$$N_8 = -0.0014 XY$$

The strain-displacement matrix can be found from $[B] = [D][N]$. $[B]$ is a 3×16 matrix.

The element stiffness matrix is given by the equation

$$[K]_{16 \times 16} = \int_{-b}^b \int_{-a}^a [B]^T [E] [B] t \, dx \, dy$$

$\begin{matrix} & & 16 \times 3 & 3 \times 3 & 3 \times 16 \end{matrix}$

From figure 2, a and b can be found as

$$a = 45$$

$$b = 6$$

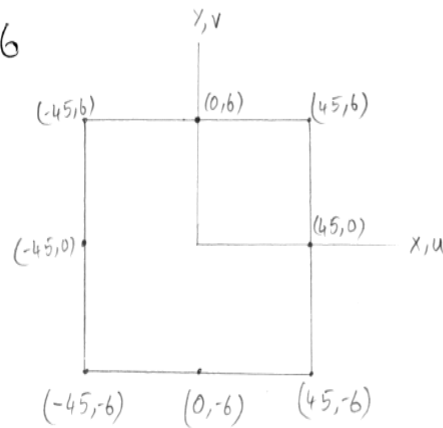


Figure-2

The local and global element stiffness matrices are given in the following pages. The displacement found from global stiffness matrix is also determined.

Local Stiffness Matrix

3245828	337302	575906	-177045	985958	4578.75	-3E+06	-79365	2E+06	138889	-7E+05	-79365	2E+06	-4579	-5E+06	-140415
337302	9171754	-140415	2E+06	-4578.8	2755800	-79365	-7E+06	138889	5E+06	-79365	-2E+06	4578.8	5E+06	-2E+05	-1E+07
575906	-140415	2694343	0	575906	140415	0	-3E+05	-7E+05	-79365	-2E+06	0	-7E+05	79365	0	317460
-177045	1808710	0	7E+06	177045	1808710	-3E+05	0	-79365	-2E+06	0	-7E+06	79365	-2E+06	317460	0
985958	-4578.75	575906	177045	3245828	-337302	-5E+06	140415	2E+06	4578.8	-7E+05	79365	2E+06	-1E+05	-3E+06	79365
4578.75	2755800	140415	2E+06	-337302	9171754	177045	-1E+07	-4579	5E+06	79365	-2E+06	-1E+05	5E+06	79365	-7E+06
-3E+06	-79365.1	0	-317460	-5E+06	177045	1E+07	0	-5E+06	-2E+05	0	317460	-3E+06	79365	5E+06	0
-79365	-7328856	-317460	0	140415	-1E+07	0	3E+07	-1E+05	-1E+07	317460	0	79365	-7E+06	0	1E+07
1622914	138889	-673586	-79365	1935287	-4578.8	-5E+06	-1E+05	3E+06	337302	575906	-177045	985958	4578.8	-3E+06	-79365
138889	4585877	-79365	-2E+06	4578.75	5498779	-2E+05	-1E+07	337302	9E+06	-1E+05	2E+06	-4579	3E+06	-79365	-7E+06
-673586	-79365.1	-2E+06	0	-673586	79365.1	0	317460	575906	-1E+05	3E+06	0	575906	140415	0	-317460
-79365	-1842898	0	-7E+06	79365.1	-2E+06	317460	0	-2E+05	2E+06	0	7E+06	177045	2E+06	-3E+05	0
1935287	4578.75	-673586	79365	1622914	-138889	-3E+06	79365	985958	-4579	575906	177045	3E+06	-3E+05	-5E+06	140415
-4578.8	5498779	79365.1	-2E+06	-138889	4585877	79365	-7E+06	4578.8	3E+06	140415	2E+06	-3E+05	9E+06	177045	-1E+07
-5E+06	-177045	0	317460	-3E+06	79365.1	5E+06	0	-3E+06	-79365	0	-317460	-5E+06	177045	1E+07	0
-140415	-1.5E+07	317460	0	79365.1	-7E+06	0	1E+07	-79365	-7E+06	-3E+05	0	140415	-1E+07	0	3E+07

k=

Global Stiffness Matrix

3245828	337301.6	575906	-177045	985958	4578.75	-3E+06	-79365	1622914	138889	-673586	-79365	2E+06	-4579	-5E+06	-140415
337302	9171754	-140415	2E+06	-4578.8	2755800	-79365	-7E+06	138889	5E+06	-79365	-2E+06	4578.8	5E+06	-177045	-1E+07
575906	-140415	2694343	0	575906	140415	0	-3E+05	-673586	-79365	-2E+06	0	-7E+05	79365	0	317460
-177045	1808710	0	7E+06	177045	1808710	-3E+05	0	-79365	-2E+06	0	-7E+06	79365	-2E+06	317460	0
985958	-4578.75	575906	177045	3245828	-337302	-5E+06	140415	1935287	4578.8	-673586	79365	2E+06	-1E+05	-3E+06	79365.1
4578.75	2755800	140415	2E+06	-337302	9171754	177045	-1E+07	-4578.8	5E+06	79365	-2E+06	-1E+05	5E+06	79365.1	-7E+06
-3E+06	-79365.1	0	-317460	-5E+06	177045	1E+07	0	-5E+06	-177045	0	317460	-3E+06	79365	5095645	0
-79365	-7328856	-317460	0	140415	-1E+07	0	3E+07	-140415	-1E+07	317460	0	79365	-7E+06	0	1.5E+07
1622914	138888.9	-673586	-79365	1935287	-4578.8	-5E+06	-1E+05	3245828	337302	575906	-177045	985958	4578.8	-3E+06	-79365
138889	4585877	-79365	-2E+06	4578.75	5498779	-2E+05	-1E+07	337302	9E+06	-140415	2E+06	-4579	3E+06	-79365	-7E+06
-673586	-79365.1	-2E+06	0	-673586	79365.1	0	317460	575906	-140415	3E+06	0	575906	140415	0	-317460
-79365	-1842898	0	-7E+06	79365.1	-2E+06	317460	0	-177045	2E+06	0	7E+06	177045	2E+06	-317460	0
1935287	4578.755	-673586	79365	1622914	-138889	-3E+06	79365	985958	-4578.8	575906	177045	3E+06	-3E+05	-5E+06	140415
-4578.8	5498779	79365.1	-2E+06	-138889	4585877	79365	-7E+06	4578.75	3E+06	140415	2E+06	-3E+05	9E+06	177045	-1E+07
-5E+06	-177045	0	317460	-3E+06	79365.1	5E+06	0	-3E+06	-79365	0	-317460	-5E+06	177045	1E+07	0
-140415	-1.5E+07	317460	0	79365.1	-7E+06	0	1E+07	-79365	-7E+06	-317460	0	140415	-1E+07	0	2.9E+07

<u>Beam Displacement Comparison</u>					
	Hand-Calculations			Matlab FEA Program	
	D.O.F	Displacement		D.O.F	Displacement
	1	0		1	0
	2	0		2	0
	3	0		3	0
	4	0		4	0
	5	0		5	0
	6	0		6	0
	7	-0.046855841		7	-0.046855841
	8	-0.389799562		8	-0.389799562
	9	0.046811659		9	0.046811659
	10	-0.389789605		10	-0.389789605
	11	-0.064484258		11	-0.064484258
	12	-1.263206701		12	-1.263206701
	13	-2.56E-07		13	-2.56E-07
	14	-1.262886667		14	-1.262886667
	15	0.064485742		15	0.064485742
	16	-1.263246632		16	-1.263246632

VITA

Rameez Iqbal

Candidate for the Degree of

Master of Science

Thesis: DEVELOPMENT OF A FINITE ELEMENT PROGRAM INCORPORATING
ADVANCED ELEMENT TYPES

Major Field: Civil Engineering

Biographical:

Personal Data:

Born in Hyderabad, India.

Education:

Bachelor of Engineering from Osmania University, Hyderabad, India. (2007)

Completed the requirements for the Master of Science in Civil Engineering at
Oklahoma State University, Stillwater, Oklahoma. (2009)

Experience:

Research Assistant (Oklahoma State University)

Development of a Finite Element Program to:

- Analyze a structure in a MATLAB programming environment
- Simulate Static and Dynamic Load analysis
- Incorporate Advanced Isoparametric Elements

Teaching Assistant (Oklahoma State University)

Assisted the faculty in organizing and evaluating students in a course on
Reinforced Concrete Design

Professional Memberships:

Member of American Society of Civil Engineers (ASCE)

Member of American Concrete Institute (ACI)

Name: Rameez Iqbal

Date of Degree: July, 2009

Institution: Oklahoma State University

Location: Stillwater, Oklahoma

Title of Study: DEVELOPMENT OF A FINITE ELEMENT ANALYSIS PROGRAM
INCORPORATING ADVANCED ELEMENT TYPES

Pages in Study: 107

Candidate for the Degree of Master of Science

Major Field: Civil Engineering

Scope and Method of Study:

Finite Elements is an advanced method which is being used by engineers and researchers for the calculation of general field problems. Many software packages are available to make this field easier. Different concepts of integration are used to calculate the stiffness matrices for various elements. The primary objective of this study was the development of a finite element analysis computer program using advanced finite element formulation techniques. The FEA program is capable of determining the beam deflections when a static and dynamic loading is applied. This program is written in a computer programming language known as Matlab.

Findings and Conclusions:

A FEA program for the solution of a mesh of advanced finite elements is proposed in Matlab. This program is capable of generating displacement fields and contour plots for the structural systems. Isoparametric elements are used to solve the element stiffness matrices for all selected elements and Gaussian Quadrature is used to numerically integrate. The Gauss Elimination solution technique is used to solve for structural displacements. Structural displacements are found to be coherent for different types of elements. Maximum tip deflection indicates that the linearly interpolated elements are stiffer than the quadratic interpolated elements.

ADVISER'S APPROVAL: Dr. Jonathan S. Goode
